Machine Learning for QCD Matter:
from Inverse Problems to Generative Models

Lingxiao Wang(王凌霄) (RIKEN)

July 4, 2024
EMMI Workshop at the University of Wrocław - Aspects of Criticality II

Prog.Part.Nucl.Phys. 104084(2023);
Phys. Rev. D 103, 116023, Phys. Rev. C 106, L051901, Phys. Rev. D 107, 083028, Phys. Rev. D 106, L051502;
Chin. Phys. Lett. 39, 120502, Phys. Rev. D 107, 056001, JHEP05(2024)060.

DEEP-IN Working Group

[Concept](#)
[Activities](#)
[Facilitators](#)
[Members](#)
[Contact](#)

DEEP-IN SCIENCE

深入 科学

CONCEPT

“DEEP learning for INverse problems (DEEP-IN)” in Sciences Working Group

The essence of discovery in sciences has always been rooted in the reverse engineering of natural phenomena and observational data. This paradigm of deducing the underlying laws of nature from observable outcomes forms the cornerstone of our scientific inquiry. **The DEEP-IN working group is established with the recognition that the elucidation of such complex phenomena demands a fusion of physics insights and advanced deep learning methodologies.**

In response to the evolving landscape of scientific research, our objective is to integrate cutting-edge **deep learning techniques, alongside generative models and other advanced statistical learning methods**, into the toolkit of scientists.

The DEEP-IN working group at RIKEN-iTHEMS is dedicated to creating an interdisciplinary platform that harnesses the transformative potential of artificial intelligence(AI). This platform is designed to **tackle inverse problems that span a diverse spectrum of sciences, from biology to physics and more in the future.**

iTHEMS 理化学研究所 数理創造プログラム
RIKEN Interdisciplinary Theoretical and Mathematical Sciences Program

About iTHEMS

HOME / About iTHEMS / Working Groups / DEEP-IN Working Group

DEEP-IN Working Group

“DEEP learning for INverse problems (DEEP-IN) in Sciences” working group (April 1st, 2024 -)

Lattice Computations
Gert Aarts, Swansea U.
Takumi Doi, iTHEMS
Andreas Ipp, TU Wien
Tetsuo Hatsuda, iTHEMS
Yan Lyu, iTHEMS

Now mostly physicists **->** **Future** more diverse scientists

BioPhysics: **Catherine Beauchemin,** iTHEMS
Condensed Matter Physics: **Steffen Backes,** iTHEMS
QCD Physics: **Kenji Fukushima**, UTokyo
Nuclear Physics: **Haozhao Liang**, UTokyo
Quantum Computing: **Enrico Rinaldi,** Quantinuum K.K./iTHEMS

Heavy-Ion Collisions
Long-Gang Pang, CCNU
Shuzhe Shi, THU
Kai Zhou, CUHK-ShenZhen

Astrophysics
Márcio Ferreira, Coimbra U.
Yuki Fujimoto, INT->iTHEMS
Akira Harada, NIT-Ibaraki
Zhenyu Zhu, TDLI->RIT

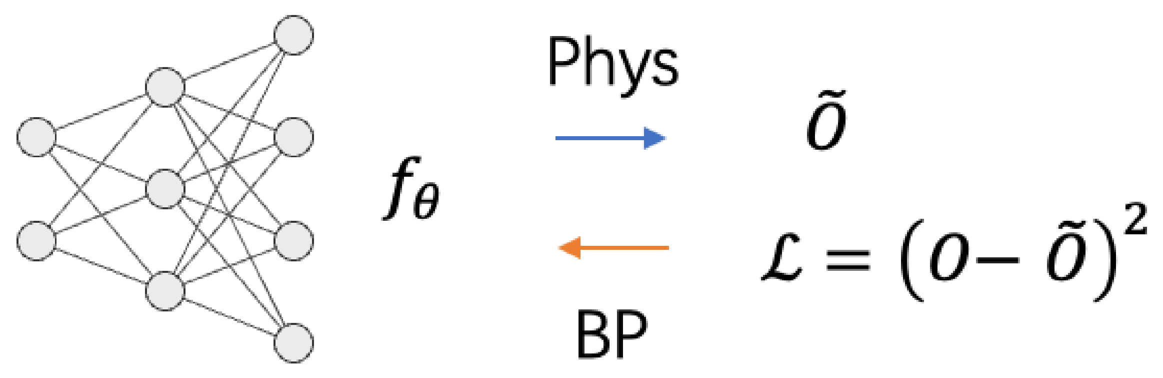
Machine Learning
Akinori Tanaka, AIP/iTHEMS
Lingxiao Wang, iTHEMS

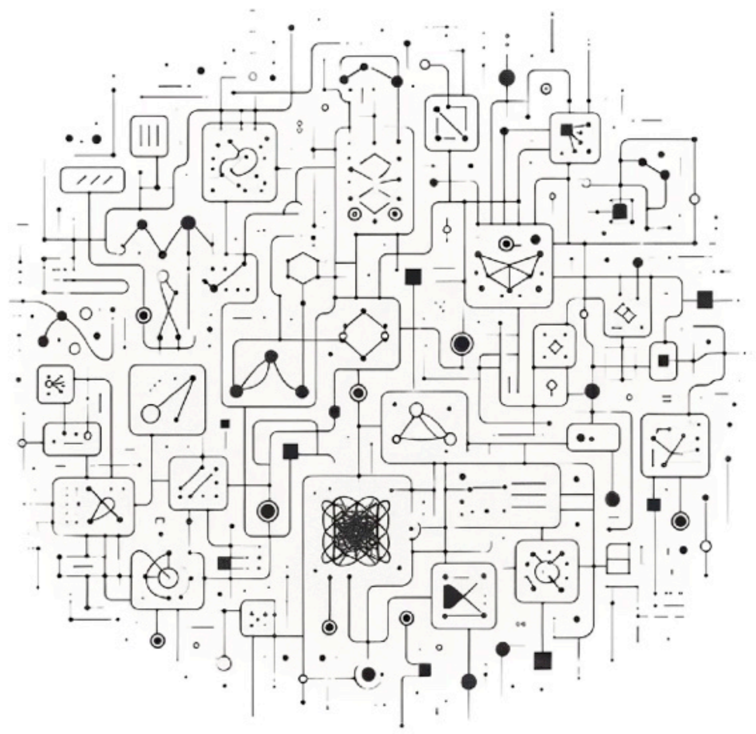
https://sites.google.com/view/deep-in-wg/homepage

Lingxiao Wang (RIKEN iTHEMS) *Contact at lingxiao.wang@riken.jp

Outline

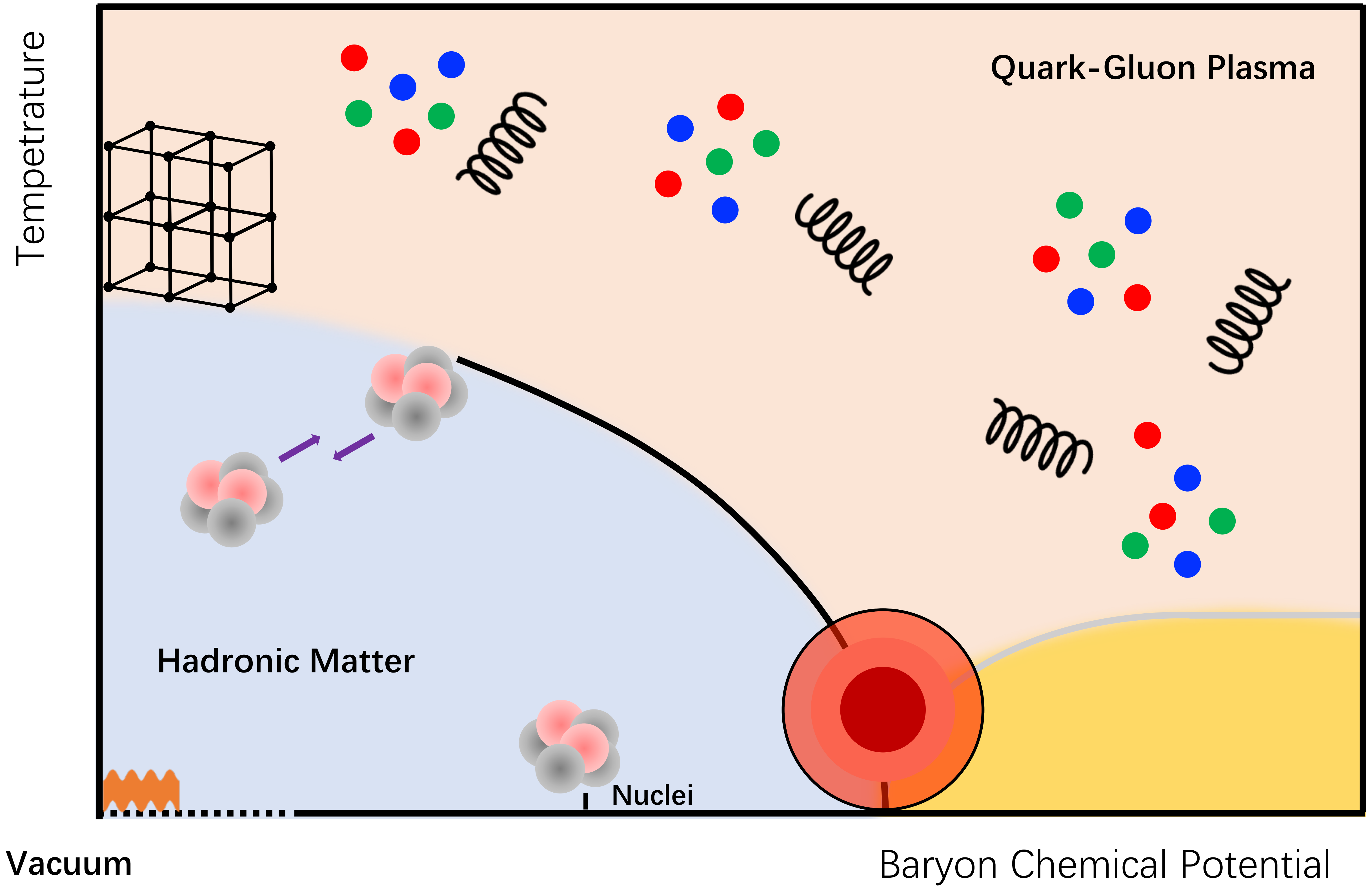
- **Machie Learning for QCD Matter**
- **Inverse Problems**
  - Data-Driven Learning
  - Physics-Driven Learning
    - Neutron Star EoSs
    - Spectral Functions
    - Hadron Forces
- **Generative Models**
- **Outlooks**





Generated by ChatGPT-4 + DALL·E

Why Machine Learning?

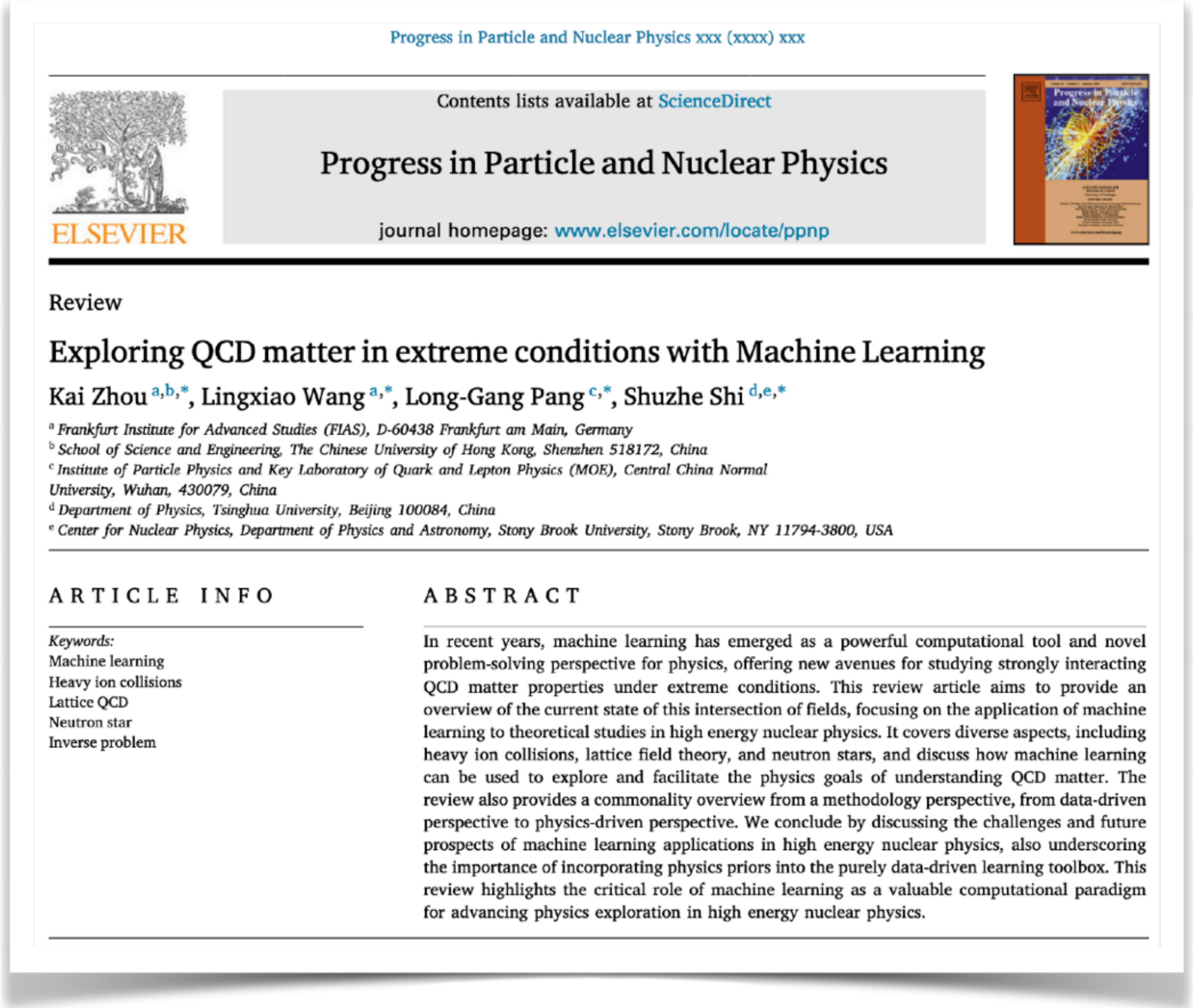


Lattice QCD
Property Extraction, Computational Effciency

Heavy-Ion Collisions
Data Analysis, Signal Detection, Fast Simulations

Neutron Stars
Data Analysis, EoS Reconstruction

Prog.Part.Nucl.Phys. 104084(2023)



Progress in Particle and Nuclear Physics xxx (xxxx) xxx

Contents lists available at ScienceDirect

Progress in Particle and Nuclear Physics

journal homepage: www.elsevier.com/locate/ppnp

ELSEVIER

Review

Exploring QCD matter in extreme conditions with Machine Learning

Kai Zhou [a,b,*], Lingxiao Wang [a,*], Long-Gang Pang [c,*], Shuzhe Shi [d,e,*]

[a] Frankfurt Institute for Advanced Studies (FIAS), D-60438 Frankfurt am Main, Germany
[b] School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen 518172, China
[c] Institute of Particle Physics and Key Laboratory of Quark and Lepton Physics (MOE), Central China Normal University, Wuhan, 430079, China
[d] Department of Physics, Tsinghua University, Beijing 100084, China
[e] Center for Nuclear Physics, Department of Physics and Astronomy, Stony Brook University, Stony Brook, NY 11794-3800, USA

ARTICLE INFO

Keywords:
Machine learning
Heavy ion collisions
Lattice QCD
Neutron star
Inverse problem

ABSTRACT

In recent years, machine learning has emerged as a powerful computational tool and novel problem-solving perspective for physics, offering new avenues for studying strongly interacting QCD matter properties under extreme conditions. This review article aims to provide an overview of the current state of this intersection of fields, focusing on the application of machine learning to theoretical studies in high energy nuclear physics. It covers diverse aspects, including heavy ion collisions, lattice field theory, and neutron stars, and discuss how machine learning can be used to explore and facilitate the physics goals of understanding QCD matter. The review also provides a commonality overview from a methodology perspective, from data-driven perspective to physics-driven perspective. We conclude by discussing the challenges and future prospects of machine learning applications in high energy nuclear physics, also underscoring the importance of incorporating physics priors into the purely data-driven learning toolbox. This review highlights the critical role of machine learning as a valuable computational paradigm for advancing physics exploration in high energy nuclear physics.

- **Heavy-Ion Collisions** : **Large number of data!** Complicated simulations!
- **Neutron Star** : **Accumulating observations!** Poor signal-noise ratio!
- **Lattice QCD** : **Computationally consuming!** Physics extraction!

What is ML?

Artificial Intelligence (AI)

Machine Learning (ML)

Deep Learning (DL)

Booming in theory and applications!

1. Big Data
2. GPU parallel
3. New architecture

2006

Geoffrey Hinton

Machine Learning (ML) is a **subset of artificial intelligence** that **involves the creation of algorithms** that **allow computers to learn from and make decisions or predictions based on data**. It's essentially a way for computers to **"learn" from data** without being explicitly programmed to do so.

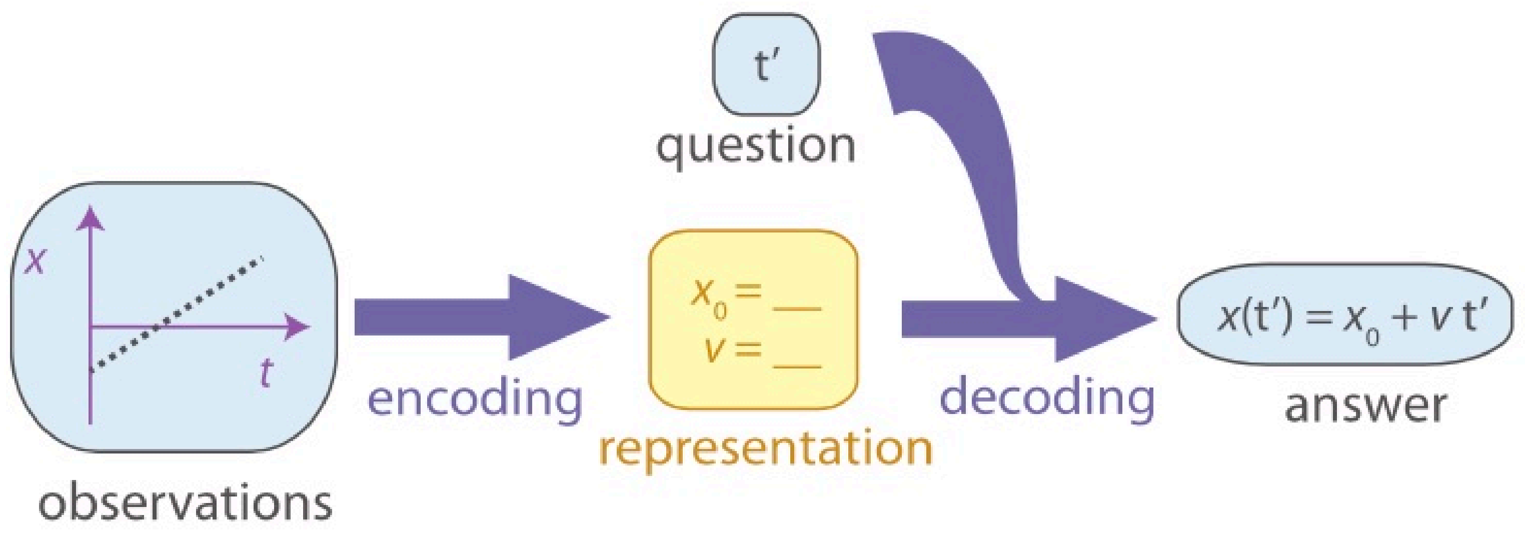
— **ChatGPT-4**

Big Data + Deep Models

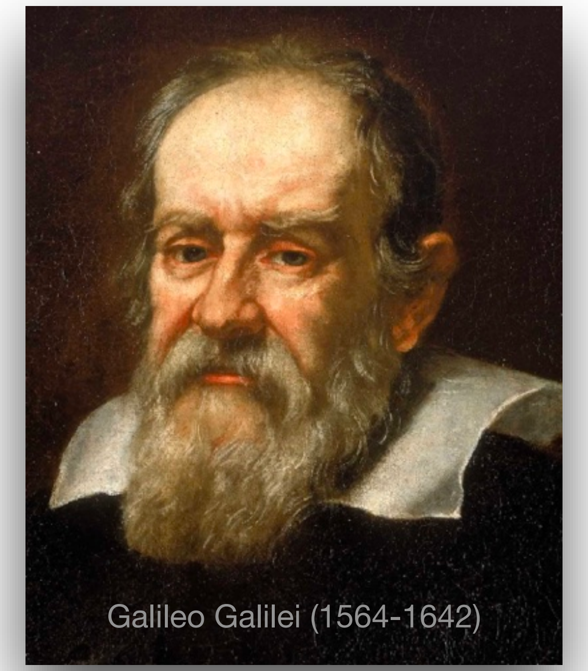
↓ **GPU** 

Successful Deep Learning!

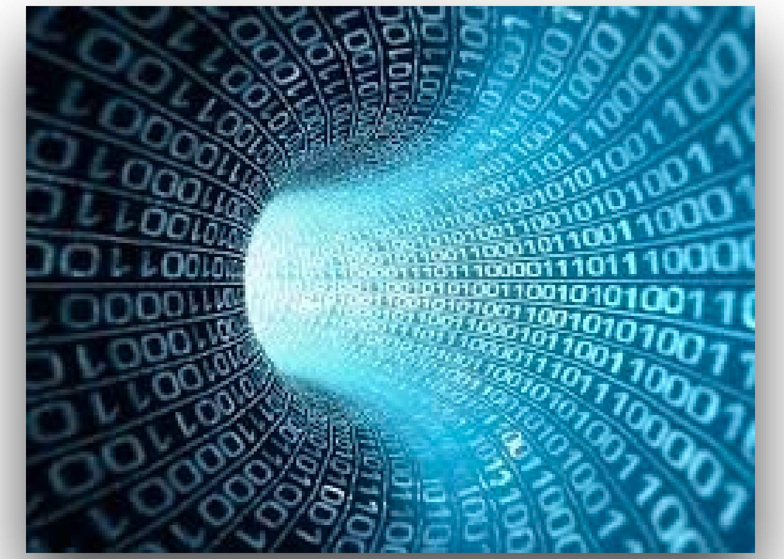
Machine Learning and Physics



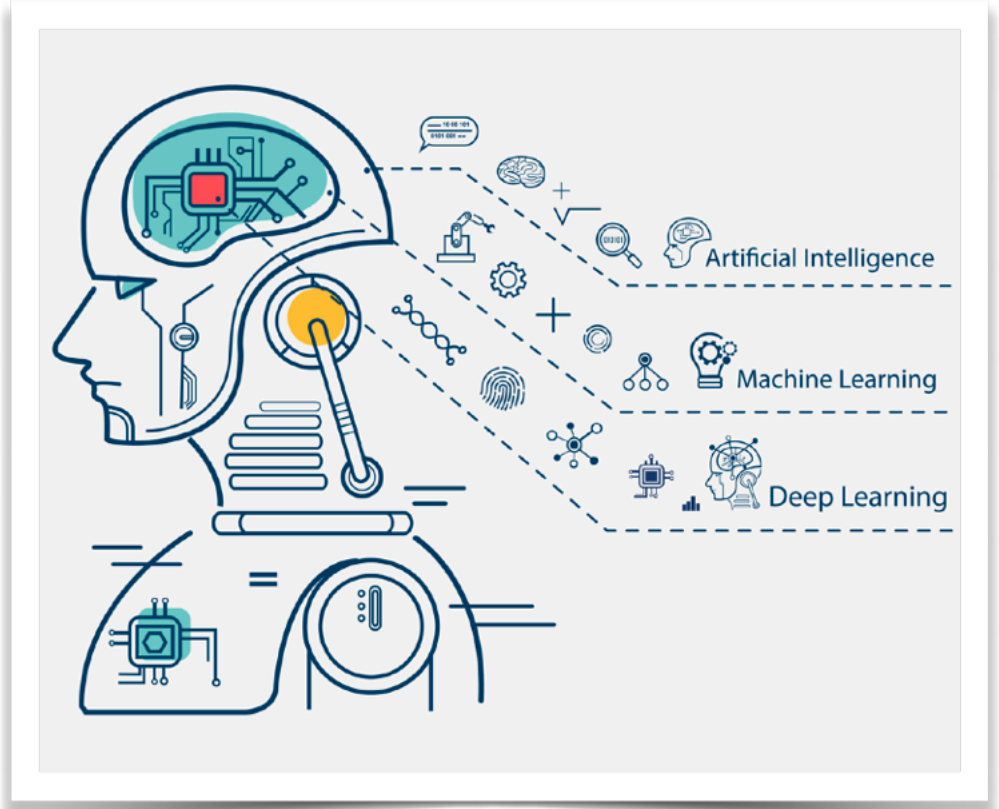
Phys.Rev. Lett. **124**, 010508 (2020)



An inverse problem in science is the process of inferring from a set of observations the causal factors that produced them.



Data, X



Machine, $\{\theta\}$

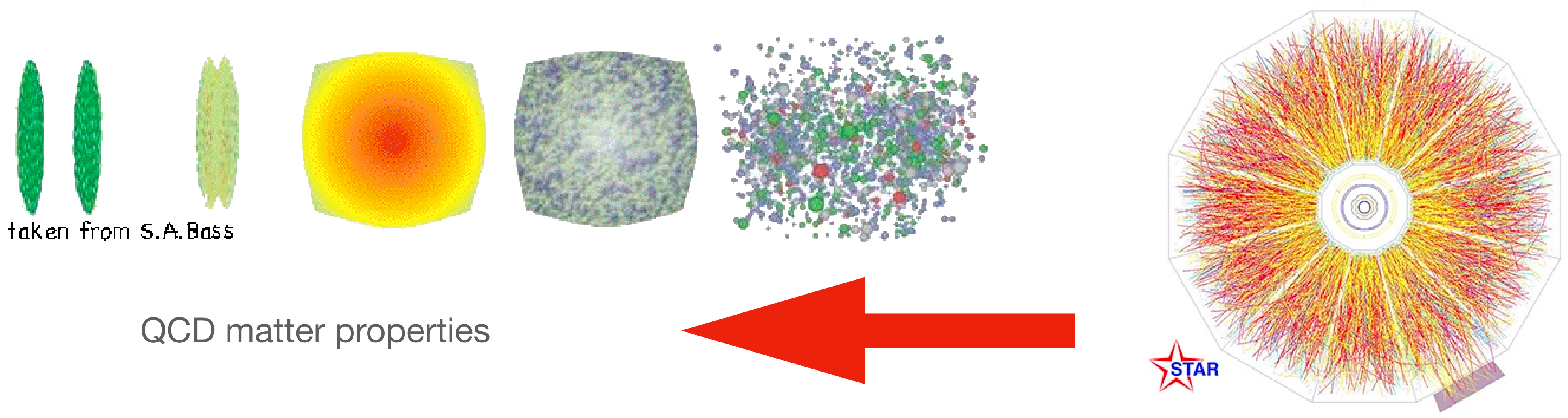
Prediction

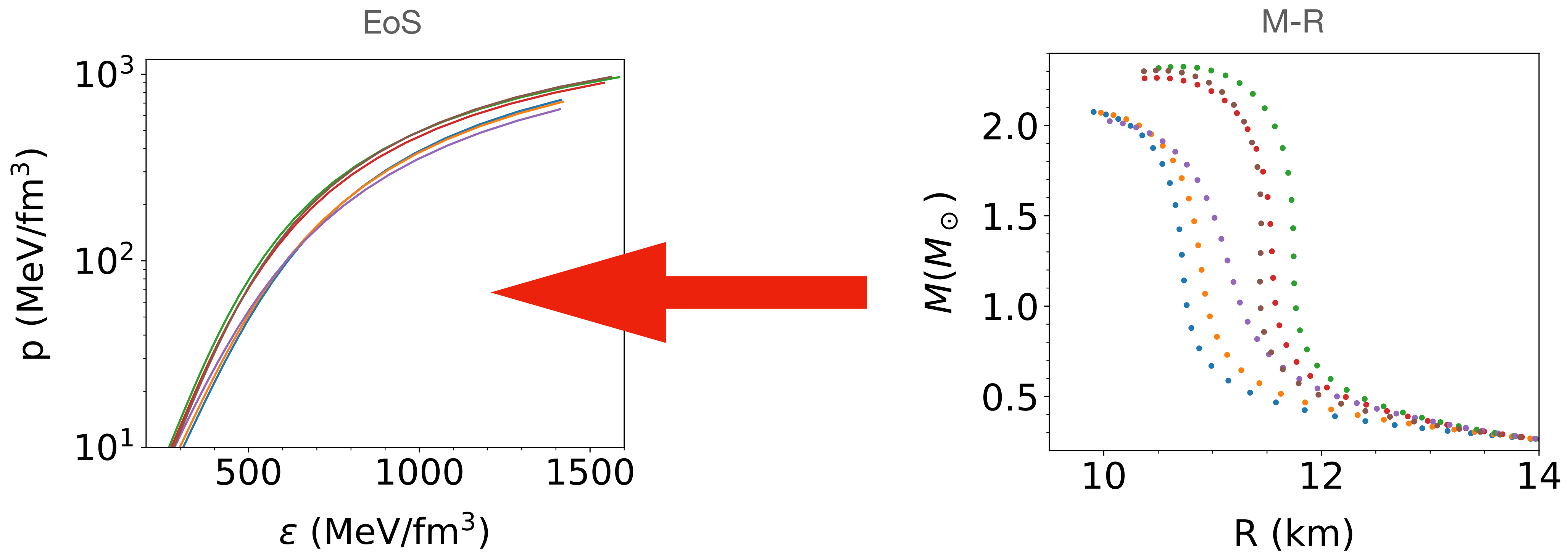
Estimation

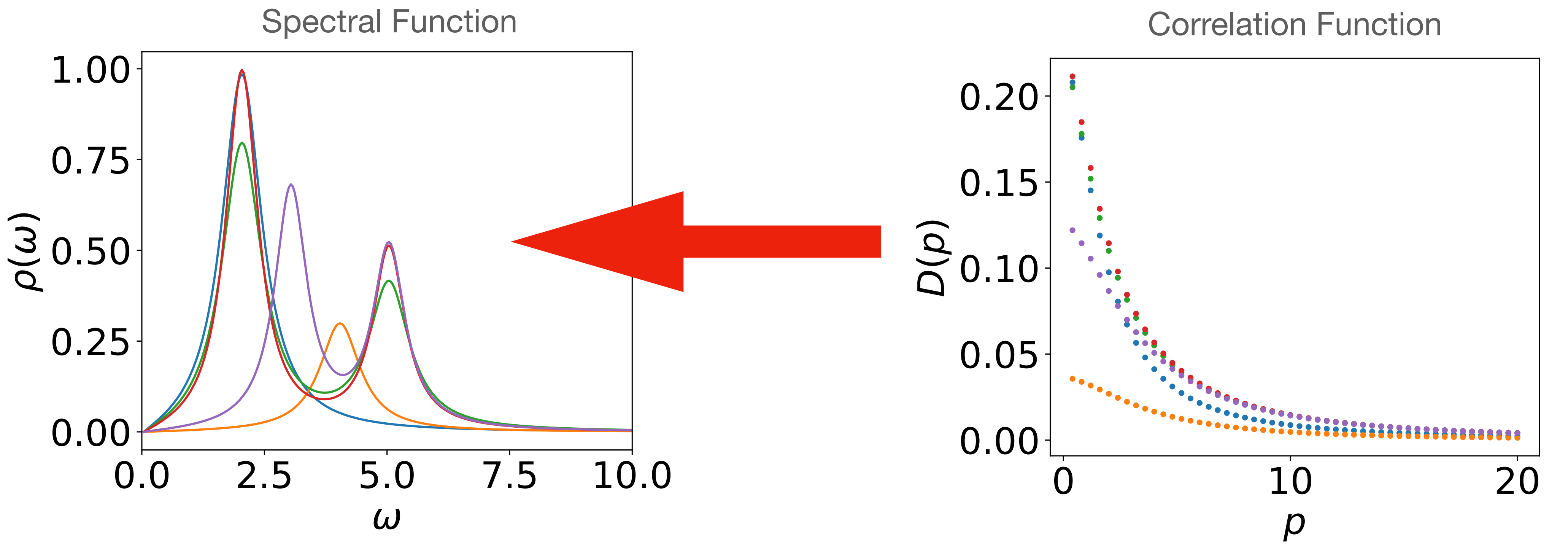
Inverse Problems

Physics

Data

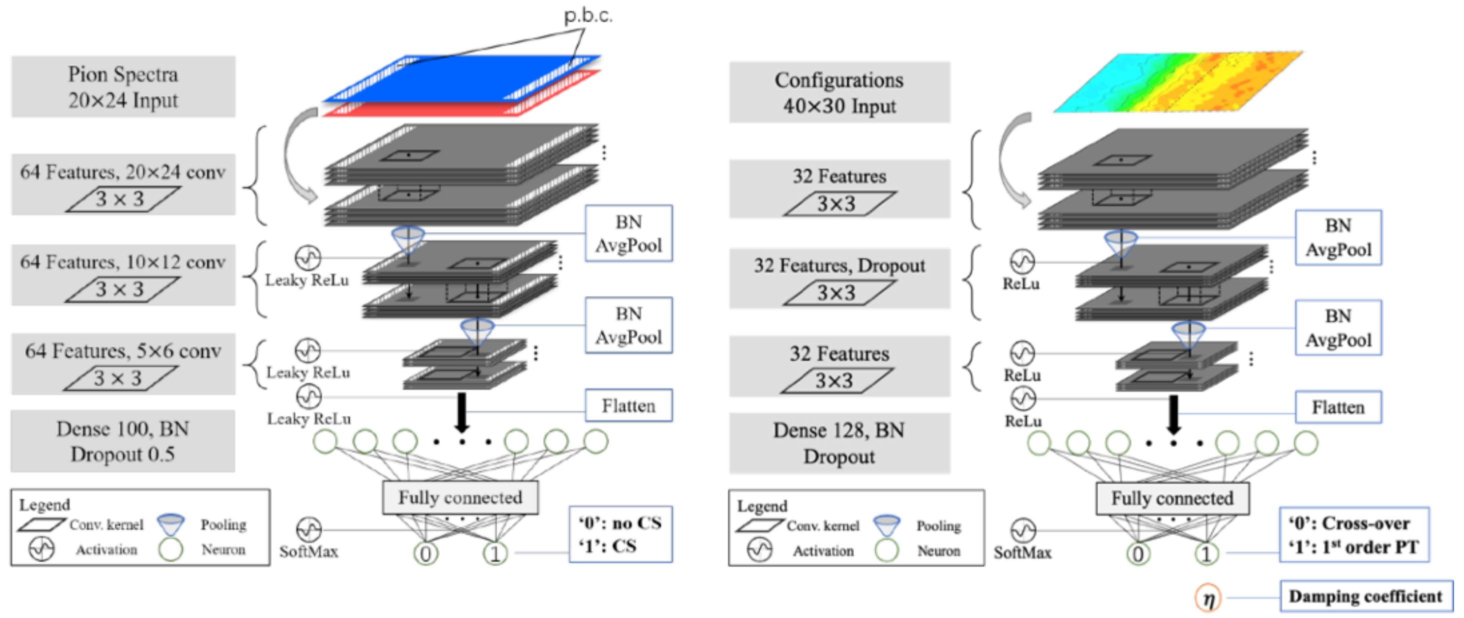






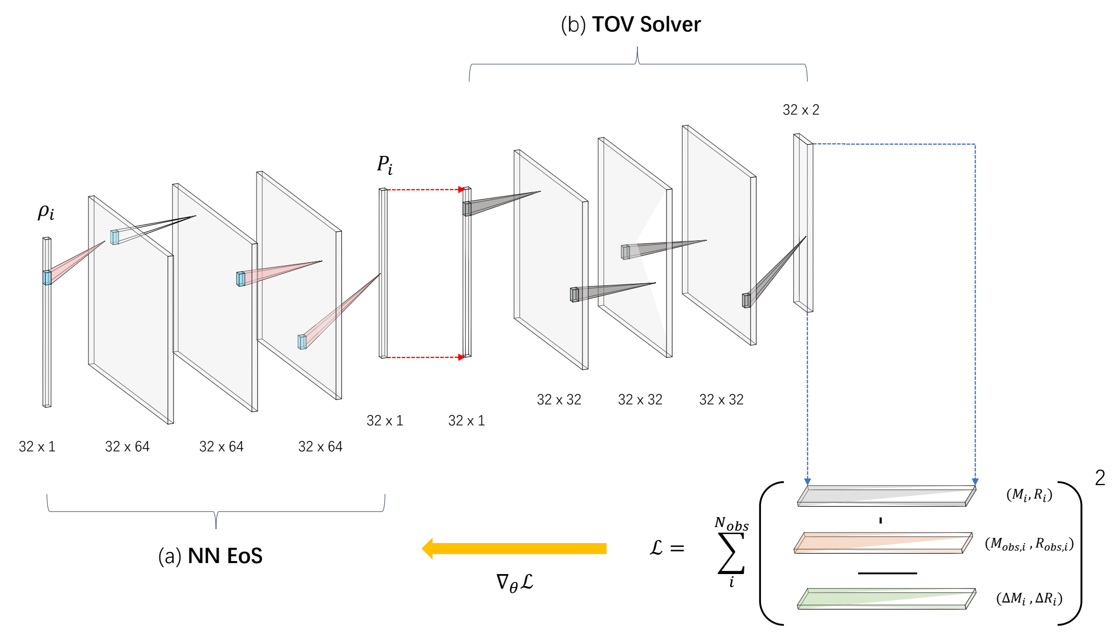
Heavy-Ion Collisions

Phys. Rev. C 106, L051901; Phys. Rev. D 103, 116023



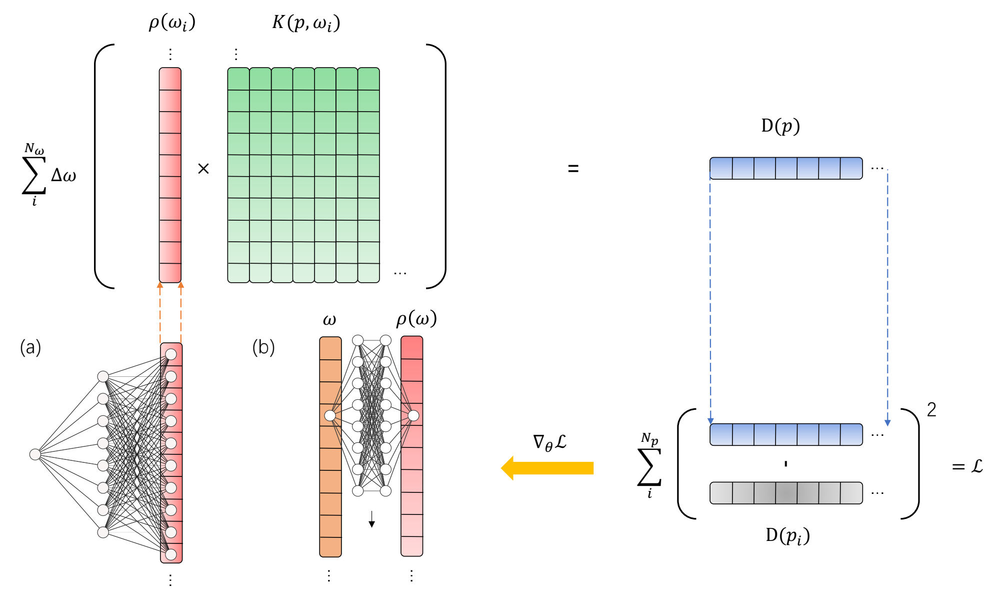
Neutron Star

Phys. Rev. D 107, 083028; JCAP08 (2022) 071



Lattice QCD

Phys. Rev. D 106, L051502; Comput. Phys. Commun. 282, 108547 (2023);



Machine Learning and Inference

Maximum Likelihood Estimation(MLE)

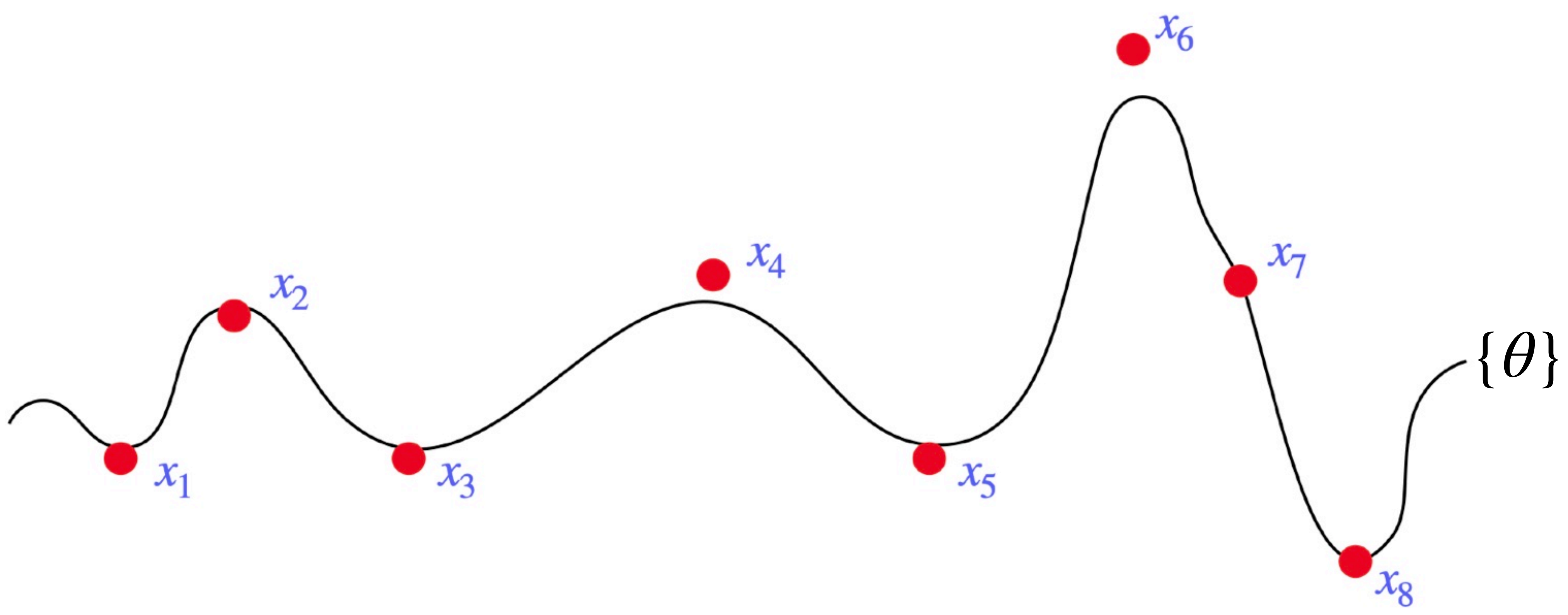
$$\max_{\theta} \prod_{i=1}^{N} p(\mathbf{x}_i \mid \theta)$$

Bayesian Inference

Maximum A Posterior(MAP)

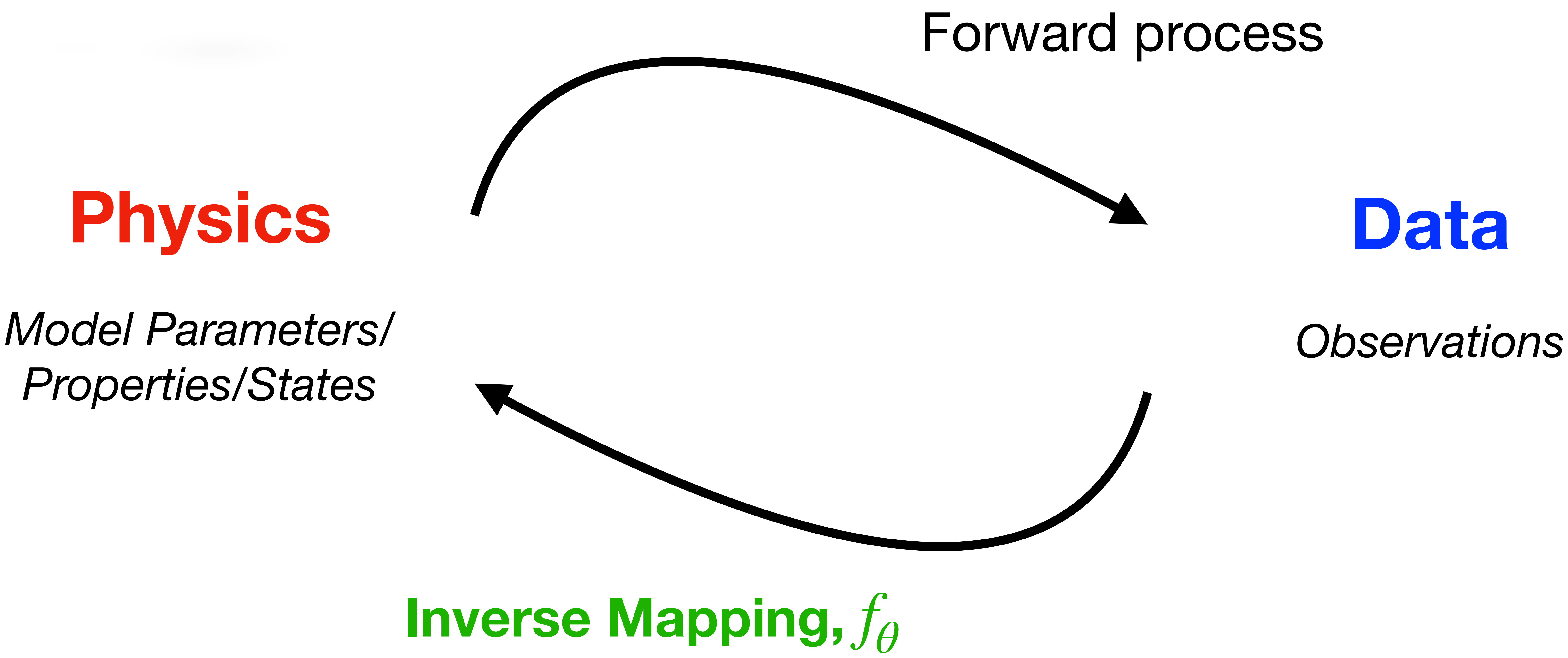
$$p(\theta \mid X) = \frac{p(X \mid \theta)\pi(\theta)}{p(X)}$$

Posterior $p(\theta \mid X)$, **Prior** $\pi(\theta)$, **Evidence** $p(X)$



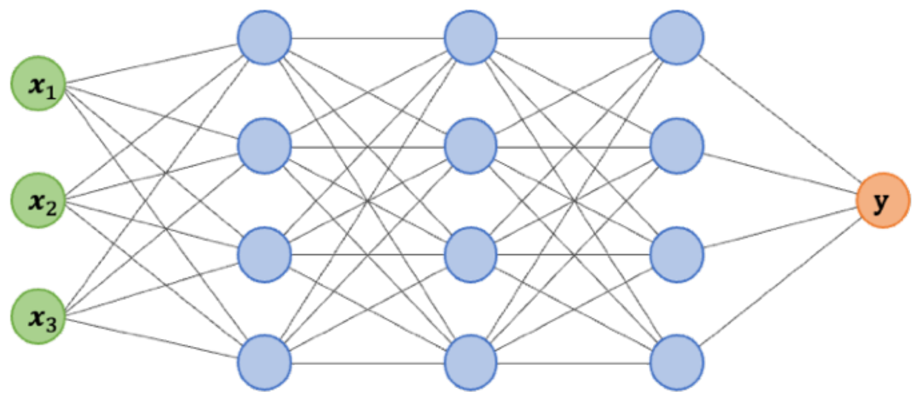
Data-Driven Learning

$$f_\theta : X \to Y$$

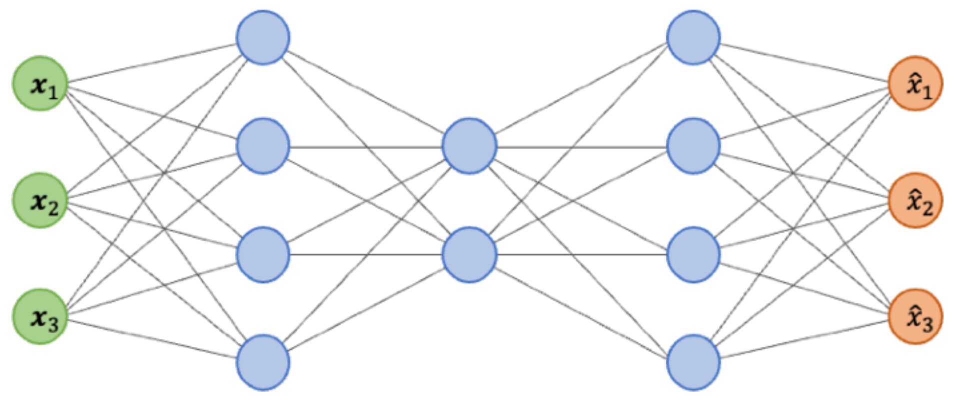


Data-Driven Learning

$$f_\theta : X \to Y$$



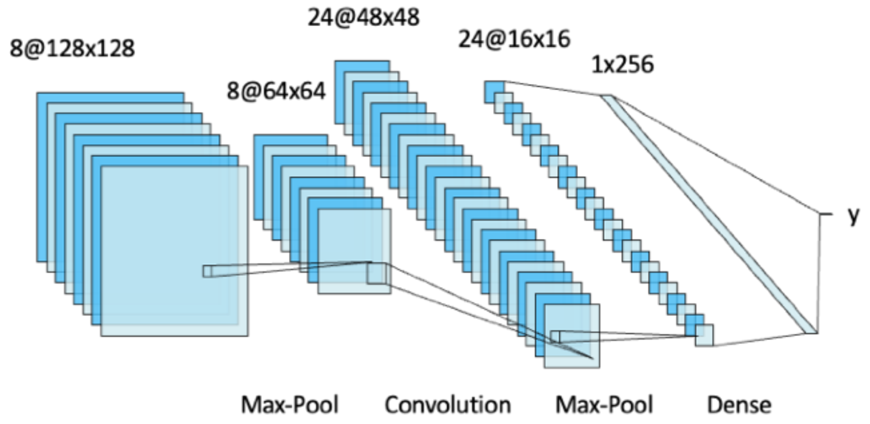
Deep Neural Network



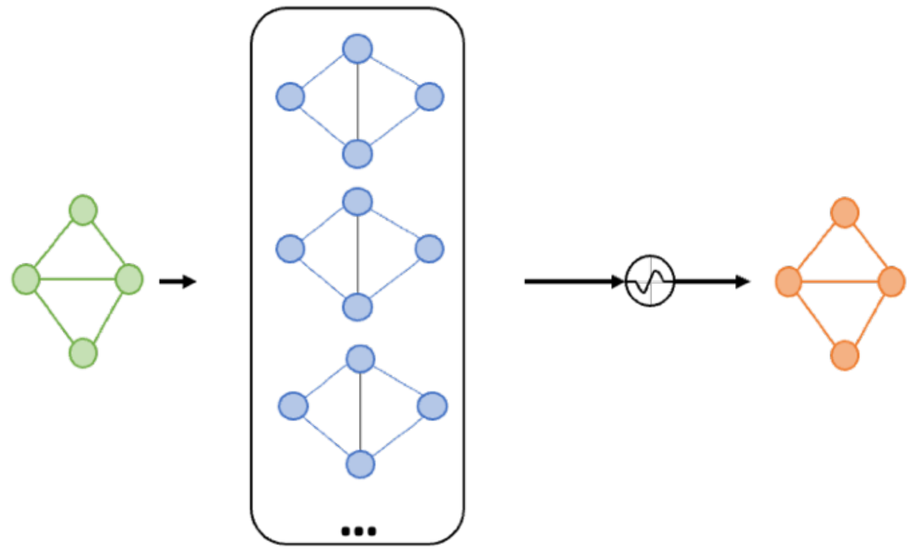
AutoEncoder

Universal Approximation Theorem (1989,1991)

A feed-forward network with *a single hidden layer* containing *a finite number of neurons* can approximate arbitrary continuous functions.



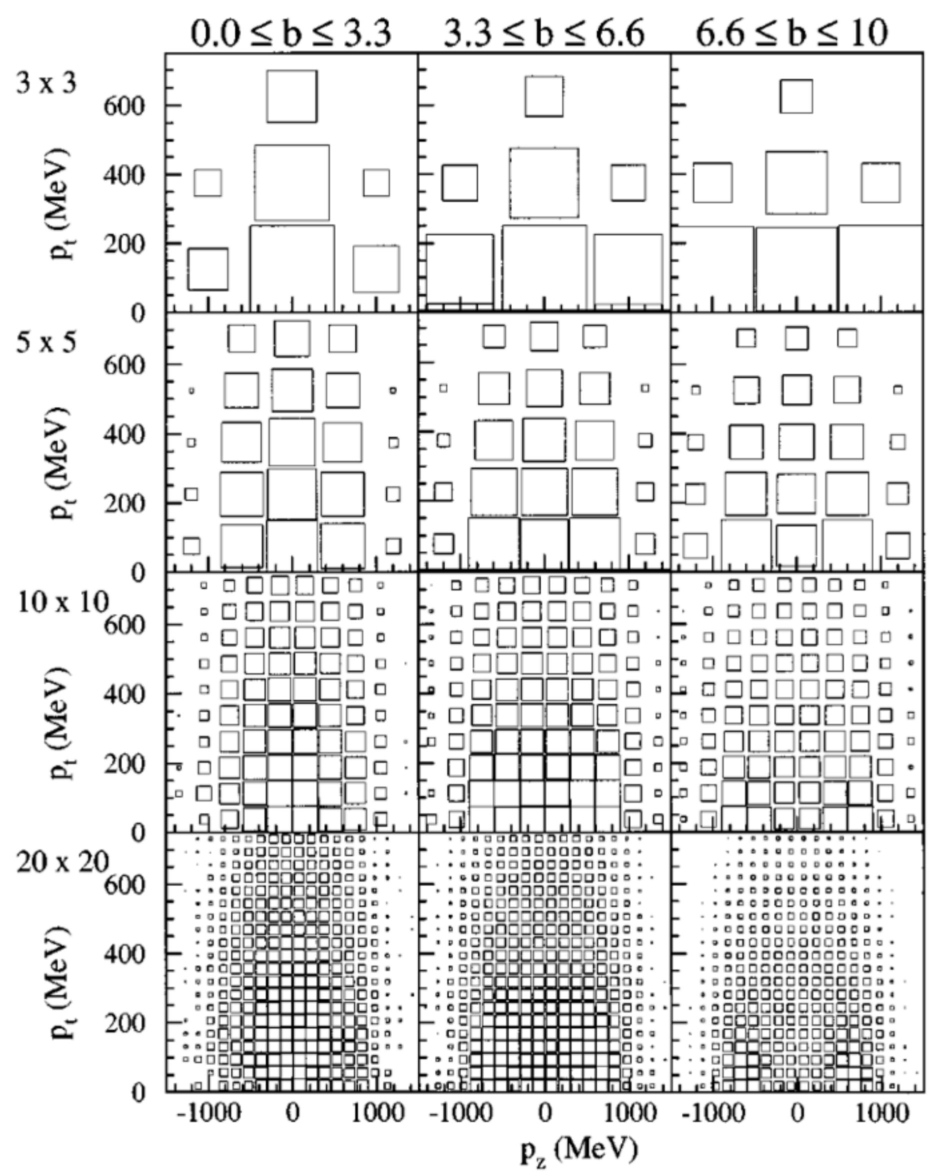
Convolutional Neural Network

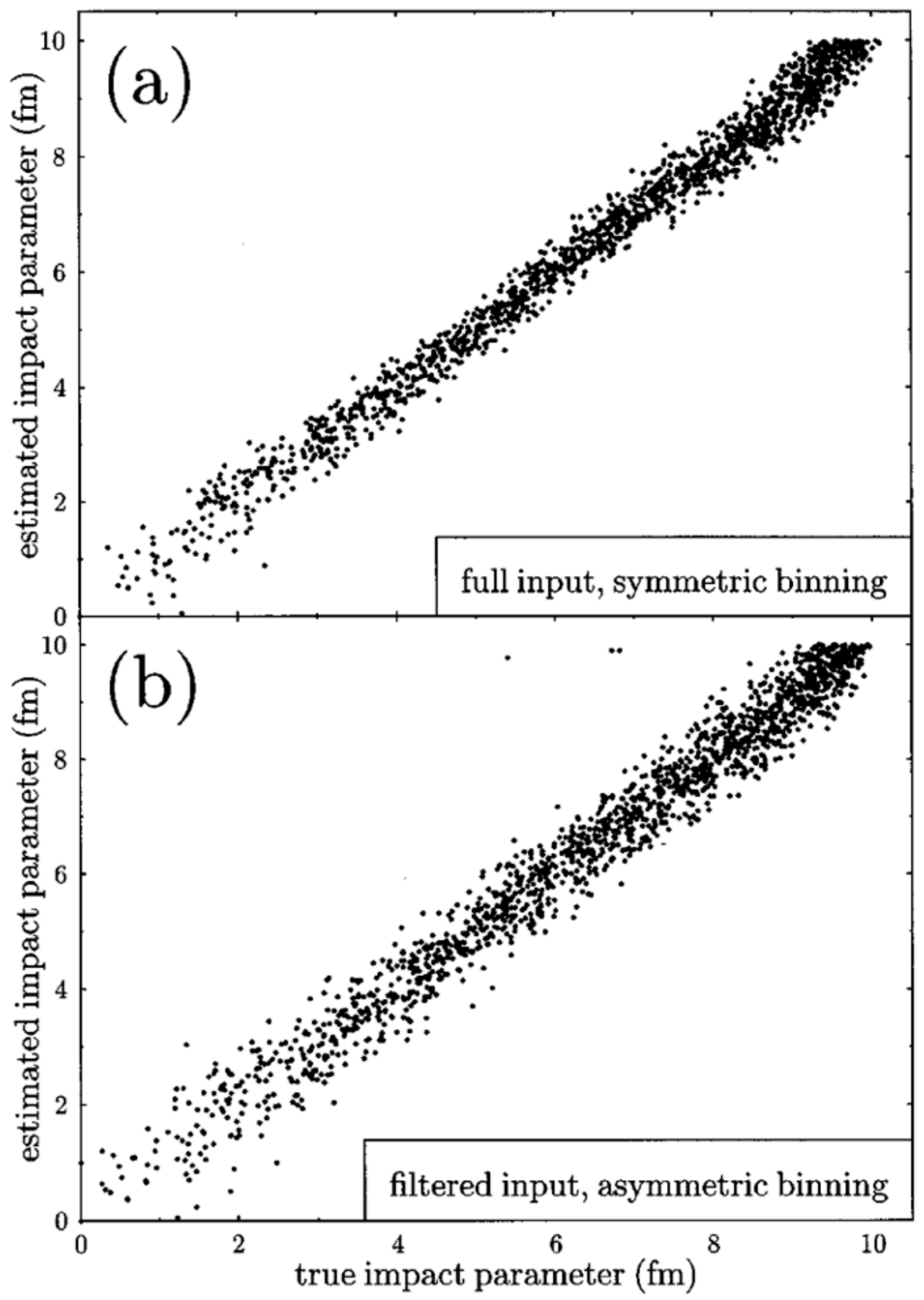


Graph Neural Network

Data-Driven Learning

Determine Impact Parameter





S. A. Bass, A. Bischoff, J. A. Maruhn, H. Stöcker, and W. Greiner, Phys. Rev. C 53, 2358 (1996)

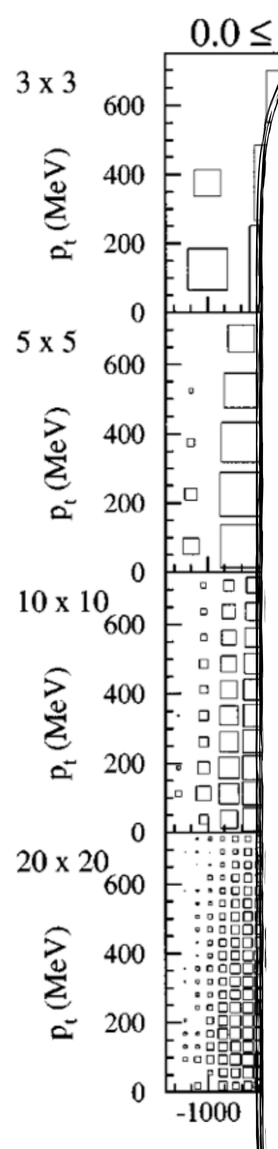
QMC data

1 hidden layer with 20 neurons

Input 5X5

Data-Driven Learning

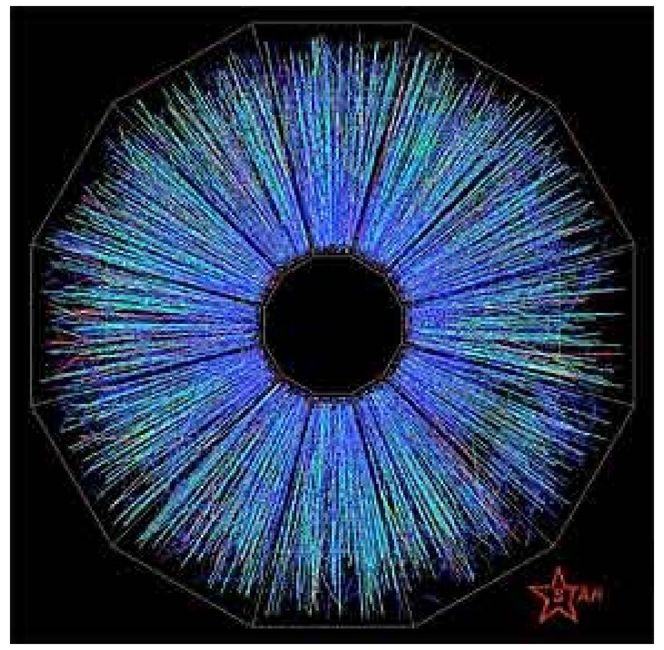
Determine Impact Parameter



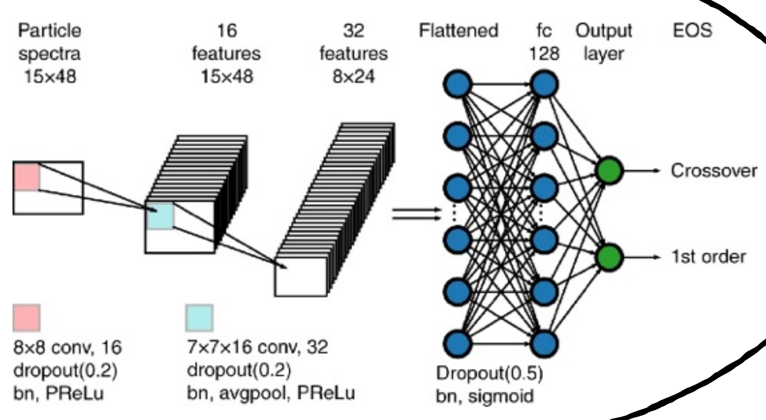
S. A

Recognizing QCD Phase Transitions

$\rho(p_T, \Phi)$



→



→

Cross-Over
or
1st order PT

L.-G. Pang, K. Zhou, N. Su, H. Petersen, H. Stöcker, and X.-N. Wang, Nature Commun. 9, 210 (2018)

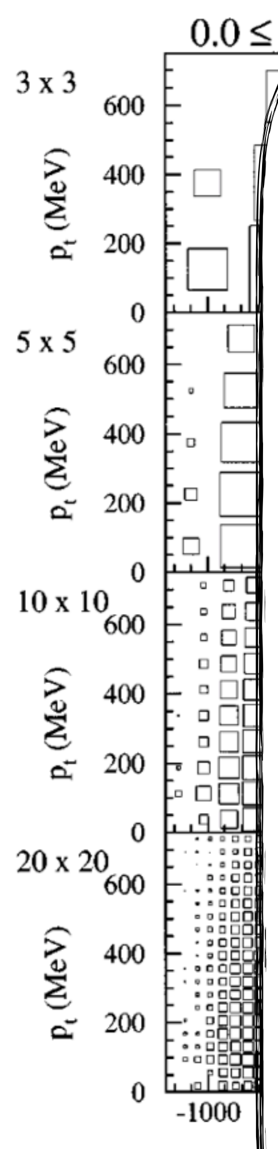
Hydro data

CNNs+DNNs

Input 15X48

Data-Driven Learning

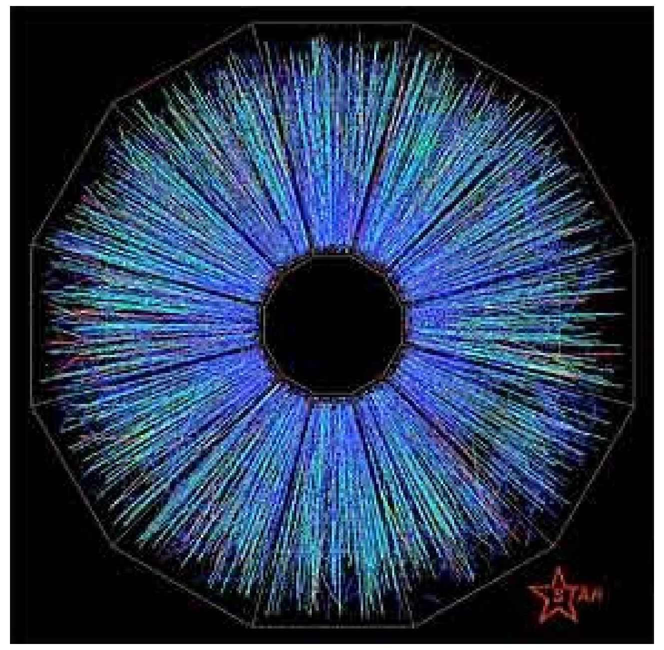
Determine Impact Parameter



S. A

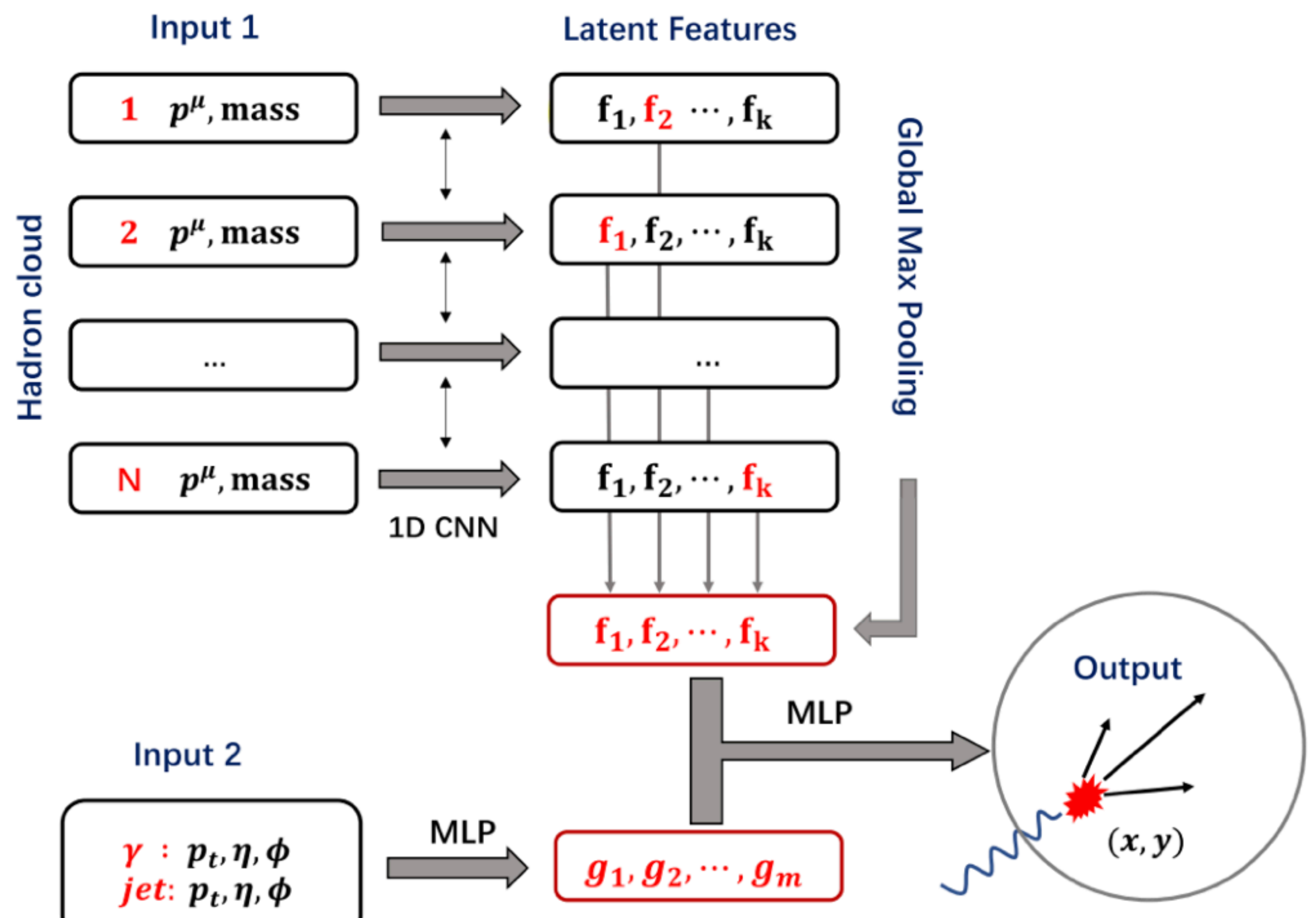
Recognizing QCD Phase Transitions

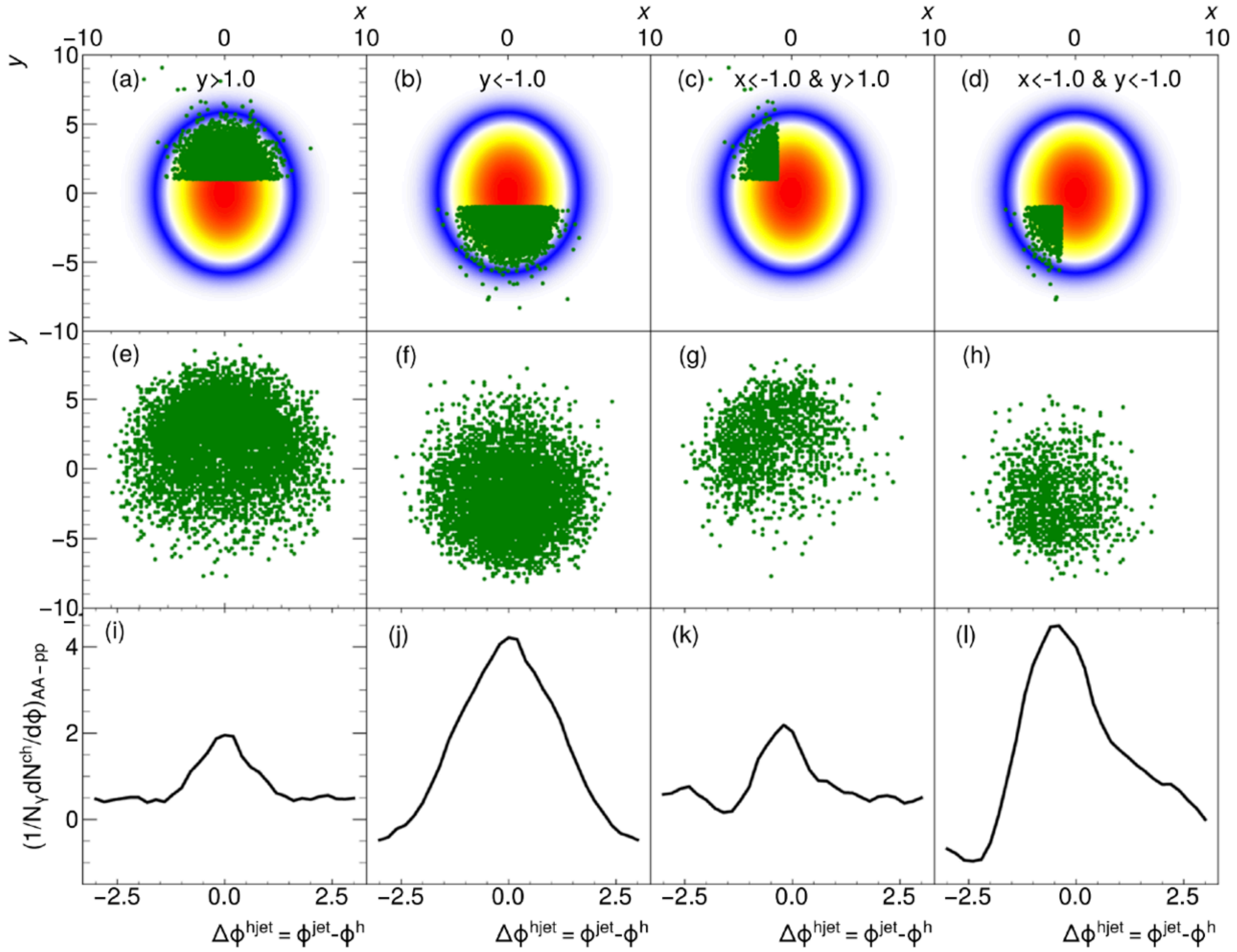
$\rho(p_T, \Phi)$



L.-G. Pang, K. Zhou, N. S

Point Net for Jet Tomography

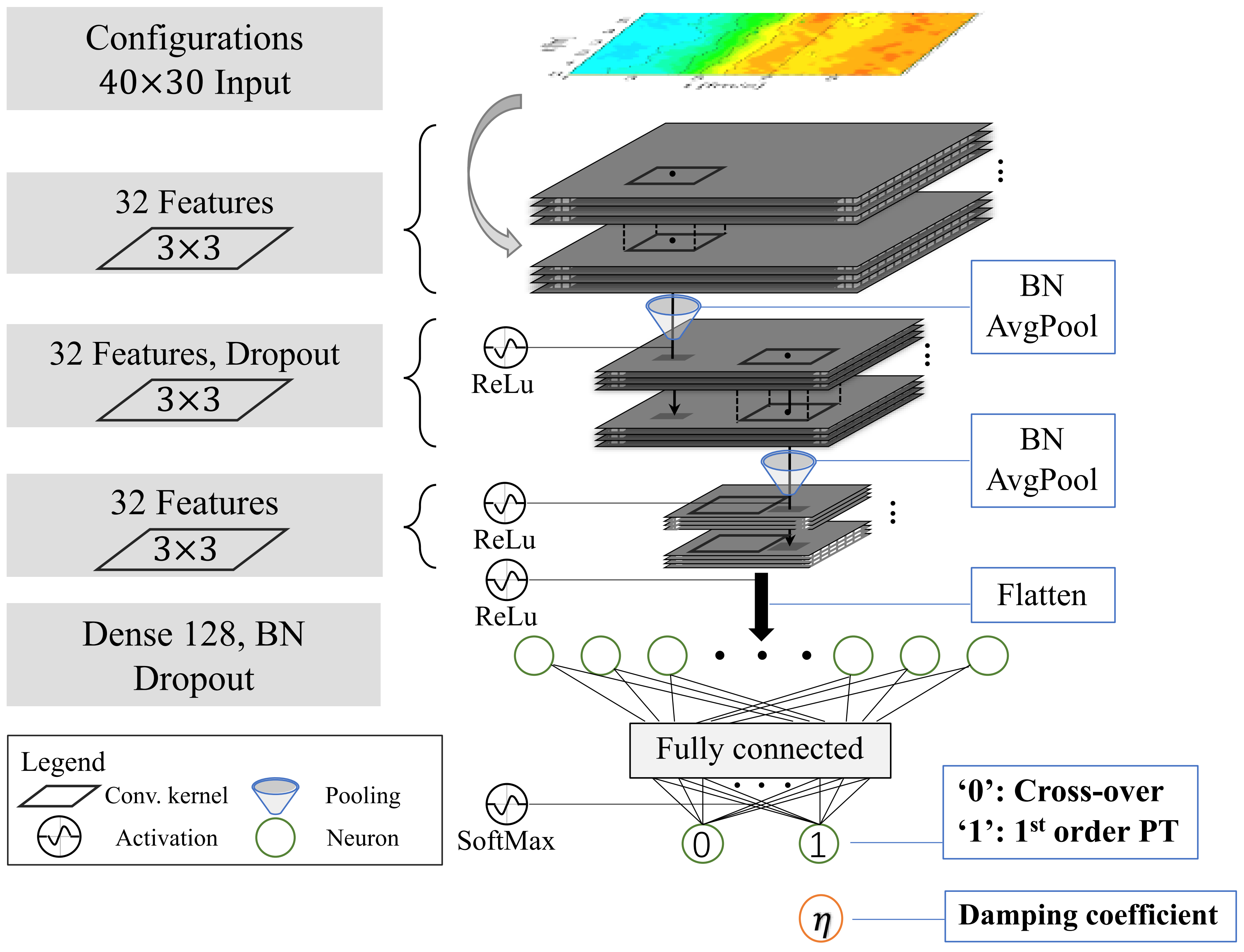




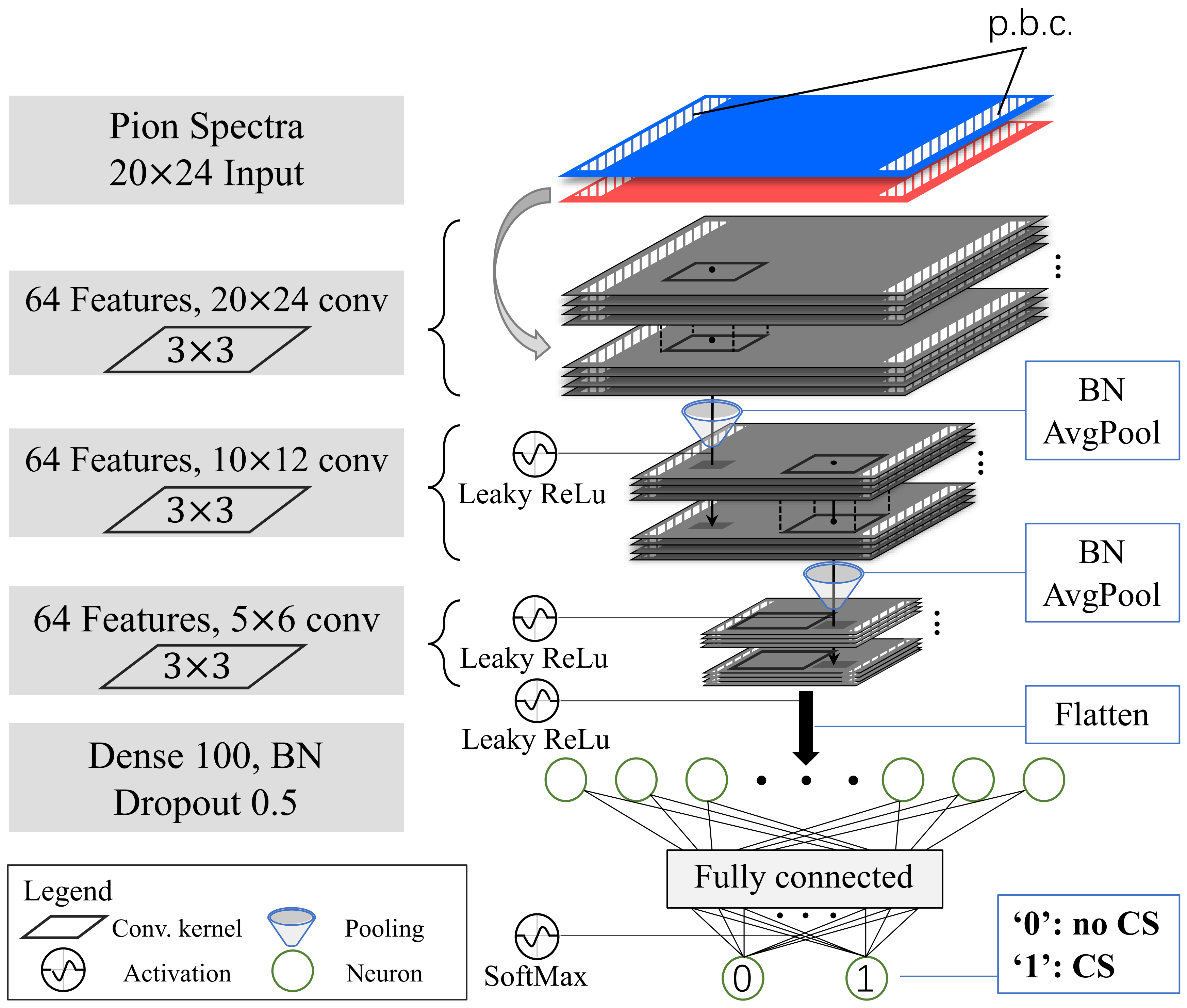
Z. Yang, Y. He, W. Chen, W.-Y. Ke, L.-G. Pang, and X.-N. Wang, Eur. Phys. J. C 83, 652 (2023).

Our Current Works

Phys. Rev. D 103, 116023



Phys. Rev. C 106, L051901(Letter)

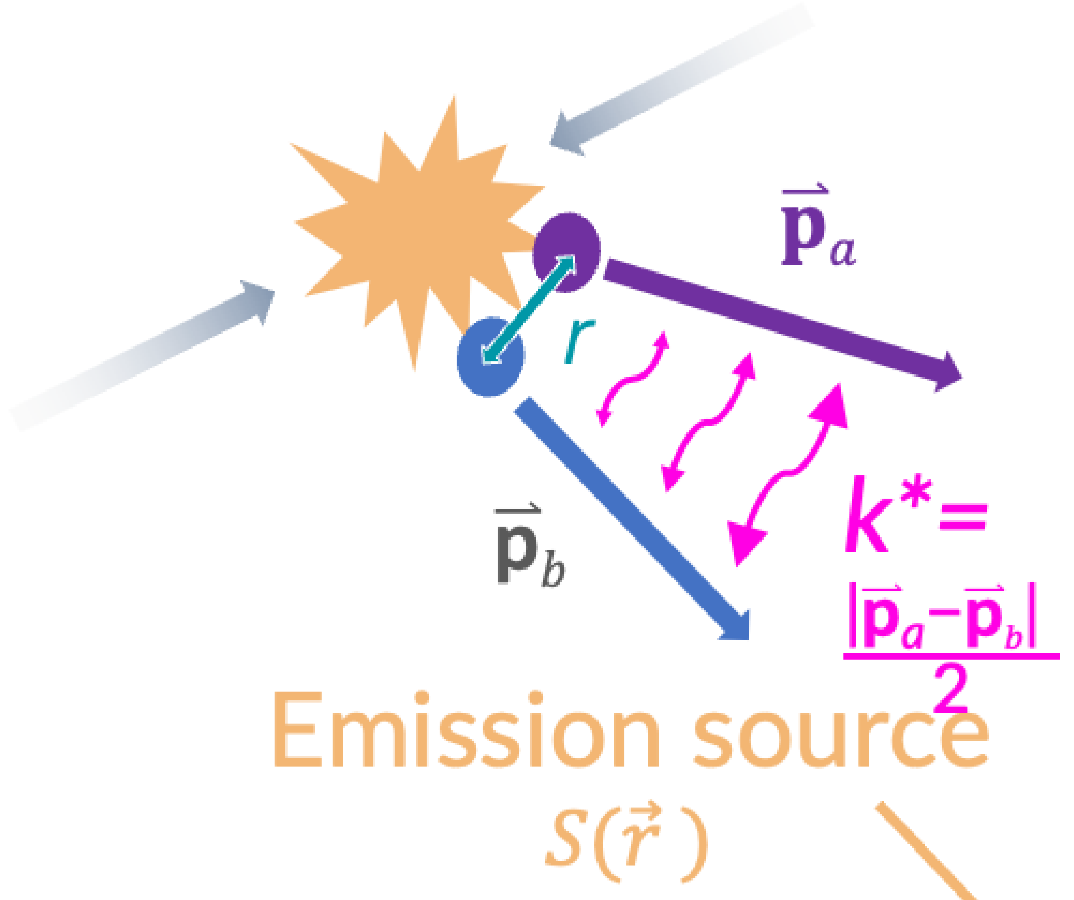


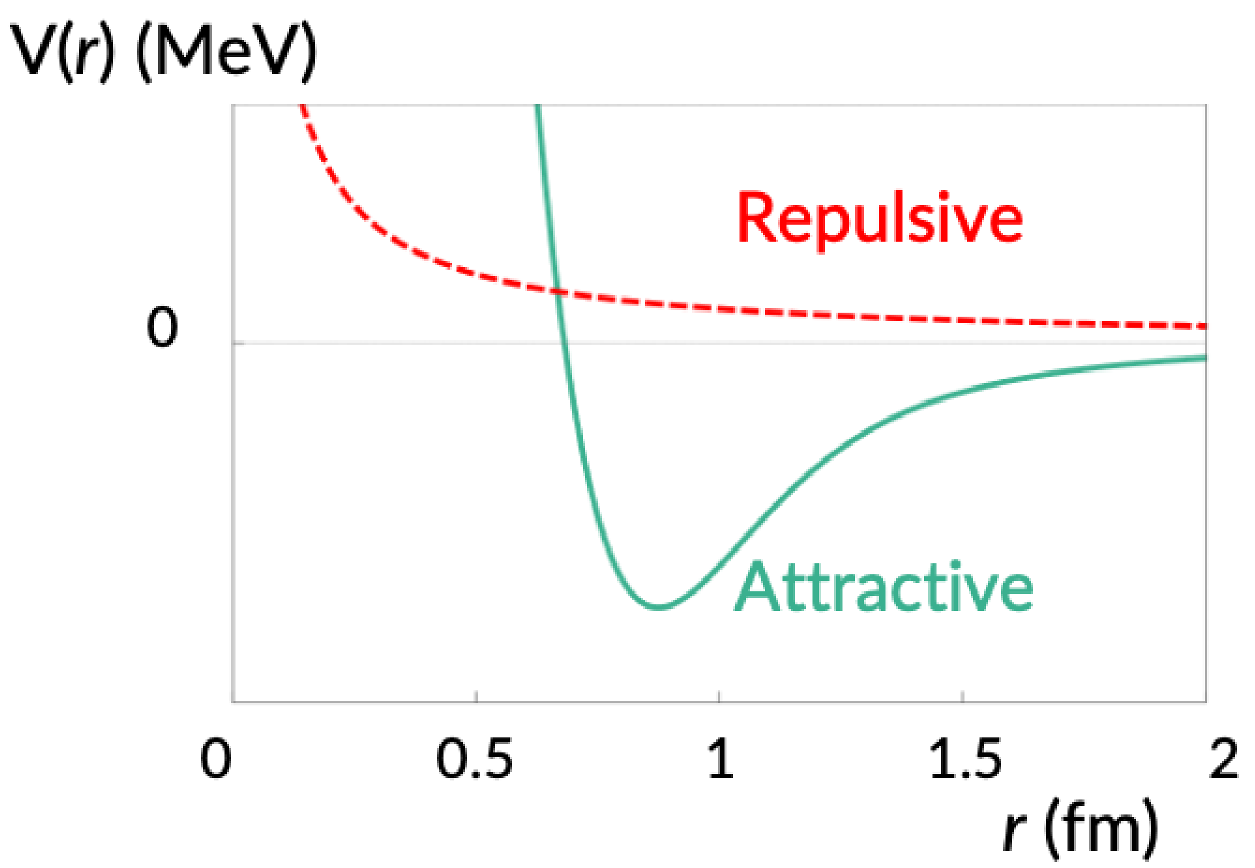
Learning phase transition orders, CME signals

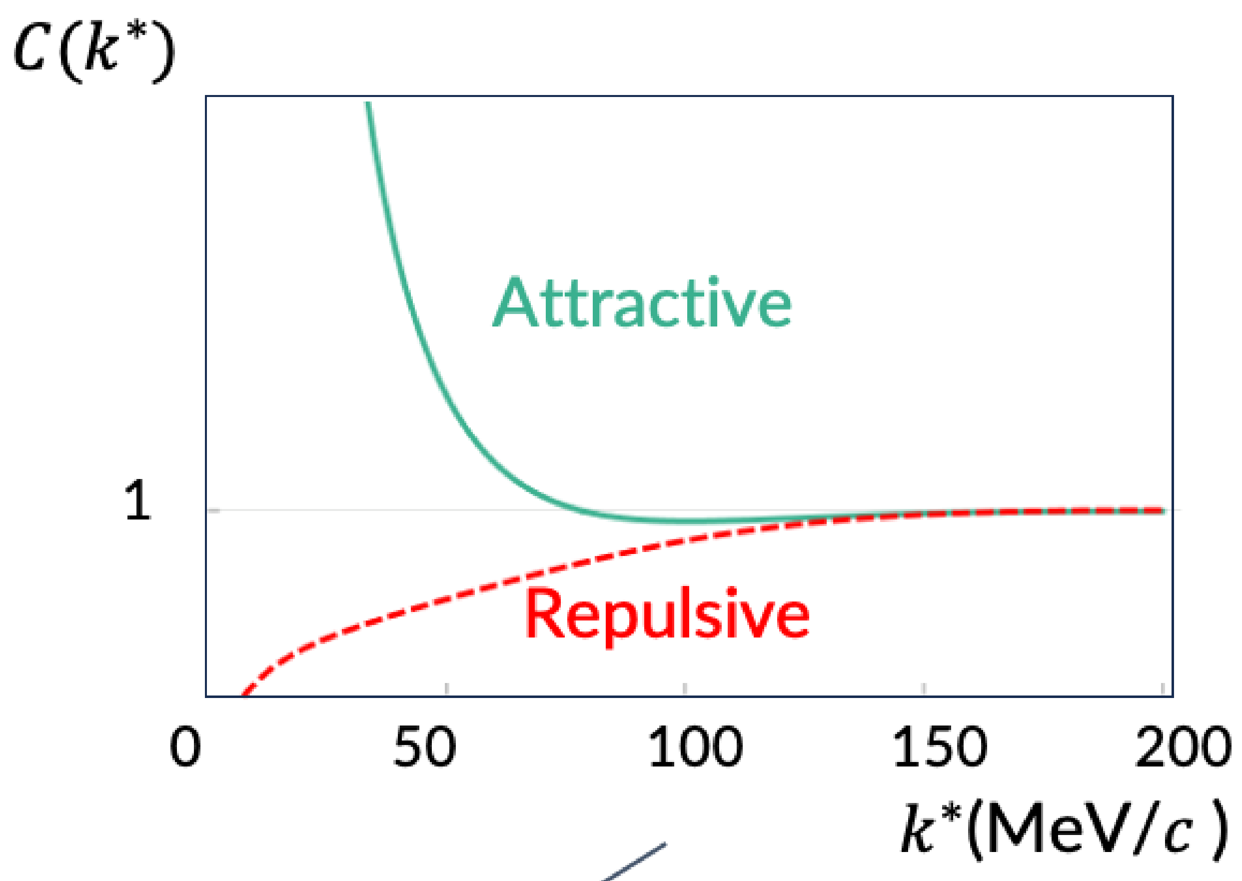
from final-stage distributions

Inverse Femtoscopy

in Preparation **with Jiaxing Zhao, Liang Zhang**







Two-particle wave function

$$C(k^*) = \int S(\vec{r}\,)\left|\psi(\vec{k}^*,\vec{r}\,)\right|^2 d^3\vec{r} \;= \mathcal{N}(k^*)\,\frac{N_{\text{same}}(k^*)}{N_{\text{mixed}}(k^*)}$$

Raffaele Del Grande | XQCD 2023

Correlations

↓ **?**

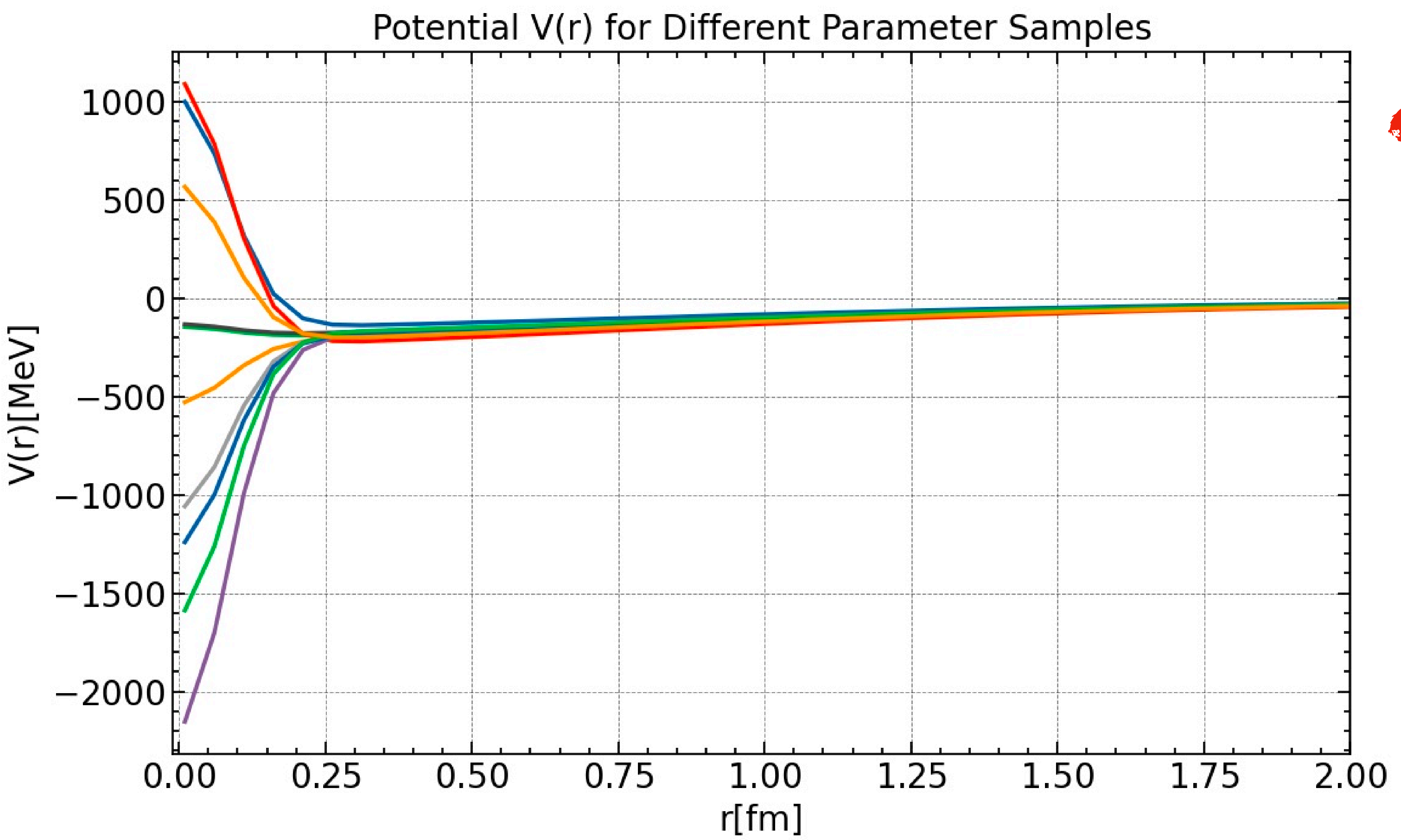
Potentials

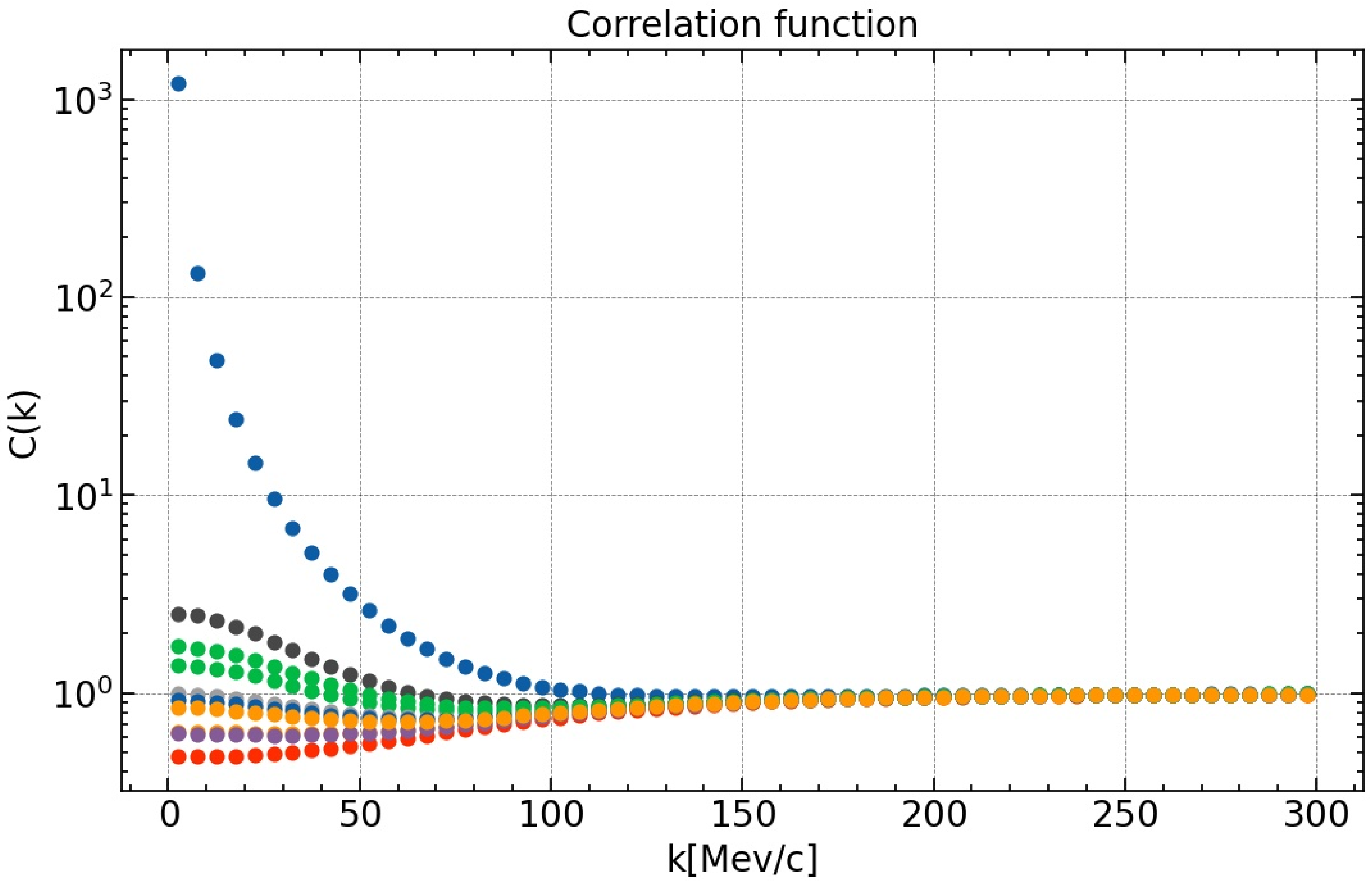
Whether this inverse mapping exists?

Inverse Femtoscopy

in Preparation **with Jiaxing Zhao, Liang Zhang**

$$V(r) = b_1 e^{-b_2 r^2} + b_3(1 - e^{-b_4 r^2})(\frac{e^{(-m_\pi r)}}{r})^{n_\pi}$$





$$S(r) = (4\pi r_0^2)^{-3/2} e^{-\frac{r^2}{4r_0^2}}$$

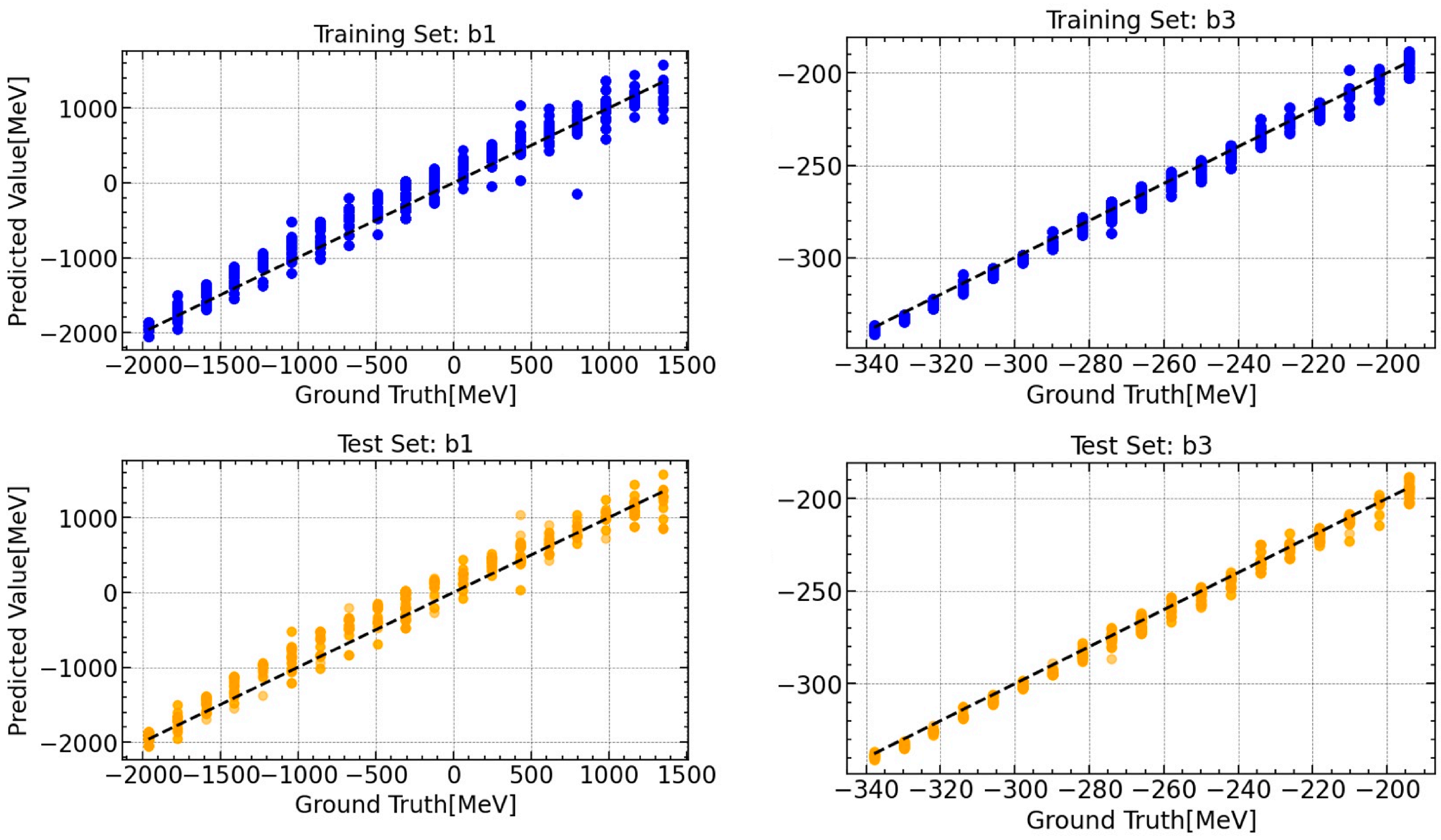
$r_0 = 1.3$ **fm**

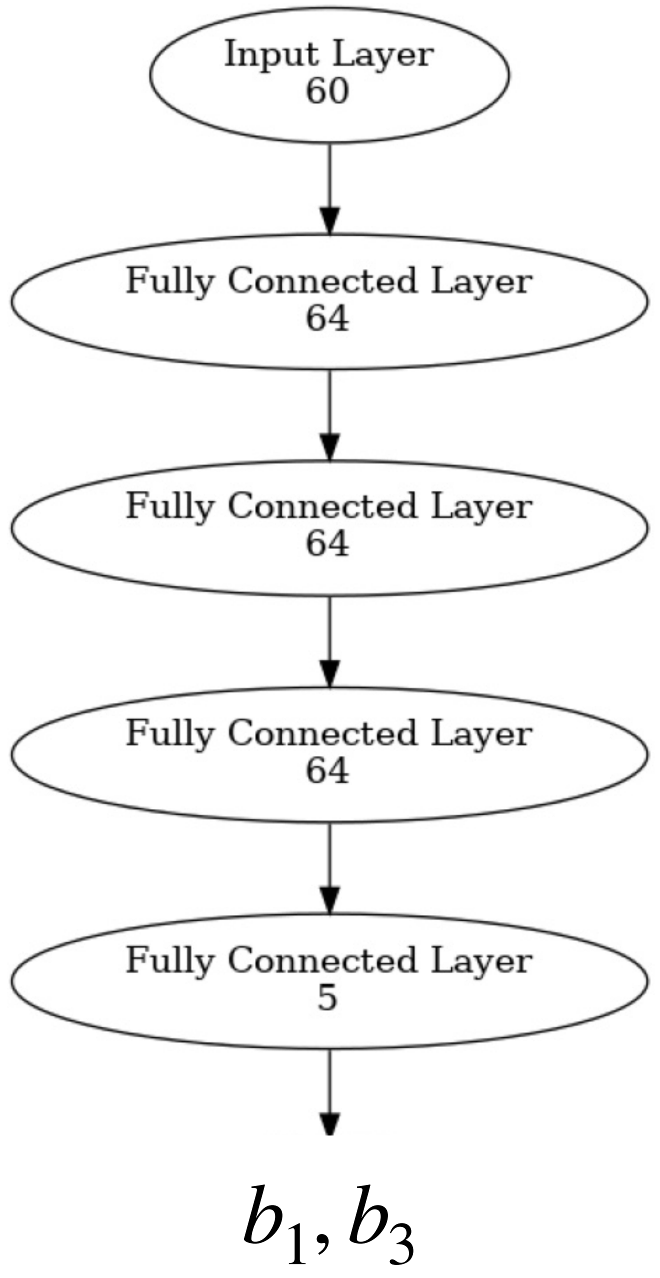
60 points(k), 10000 correlations

Inverse Femtoscopy

in Preparation

with Jiaxing Zhao, Liang Zhang





60 points

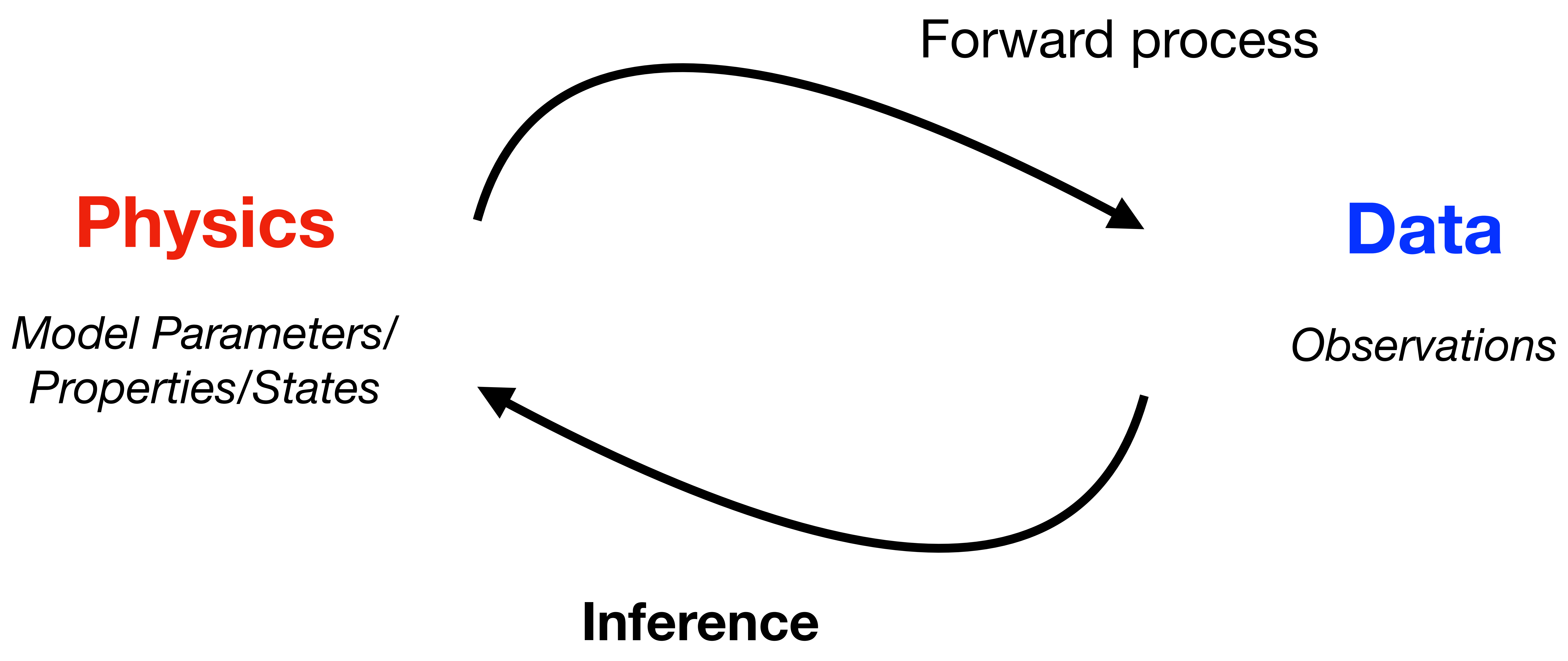
R-squared	b1	b3
Training	0.96	0.99
Testing	0.96	0.99

Neural Networks for Femtoscopy

$b_2 = 73.9, b_4 = 0.78, n_\pi = 2$

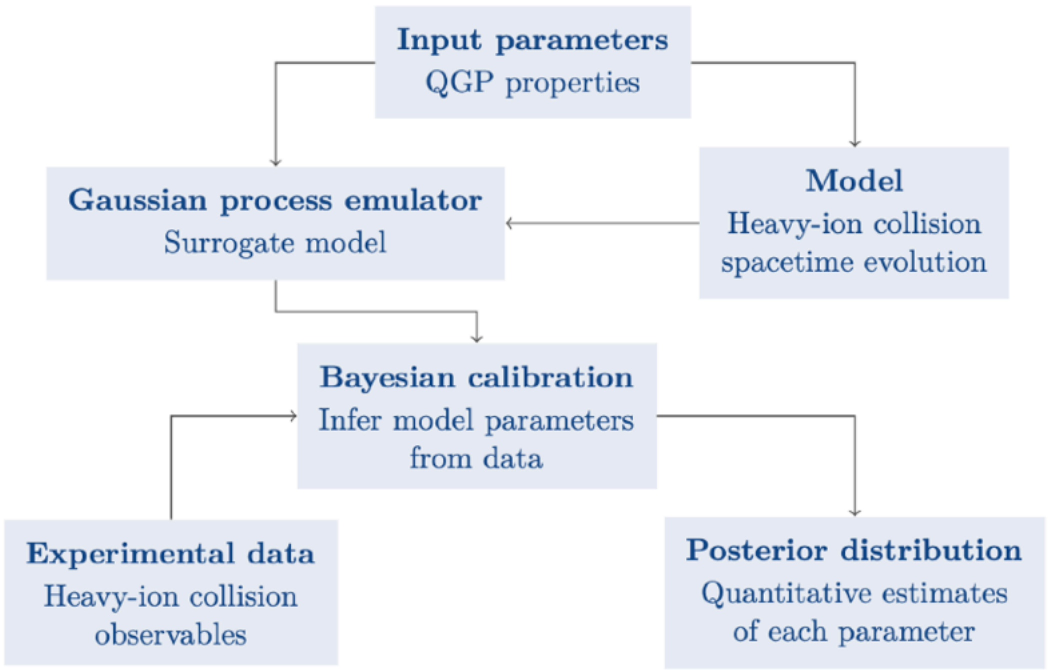
Bayesian Inference

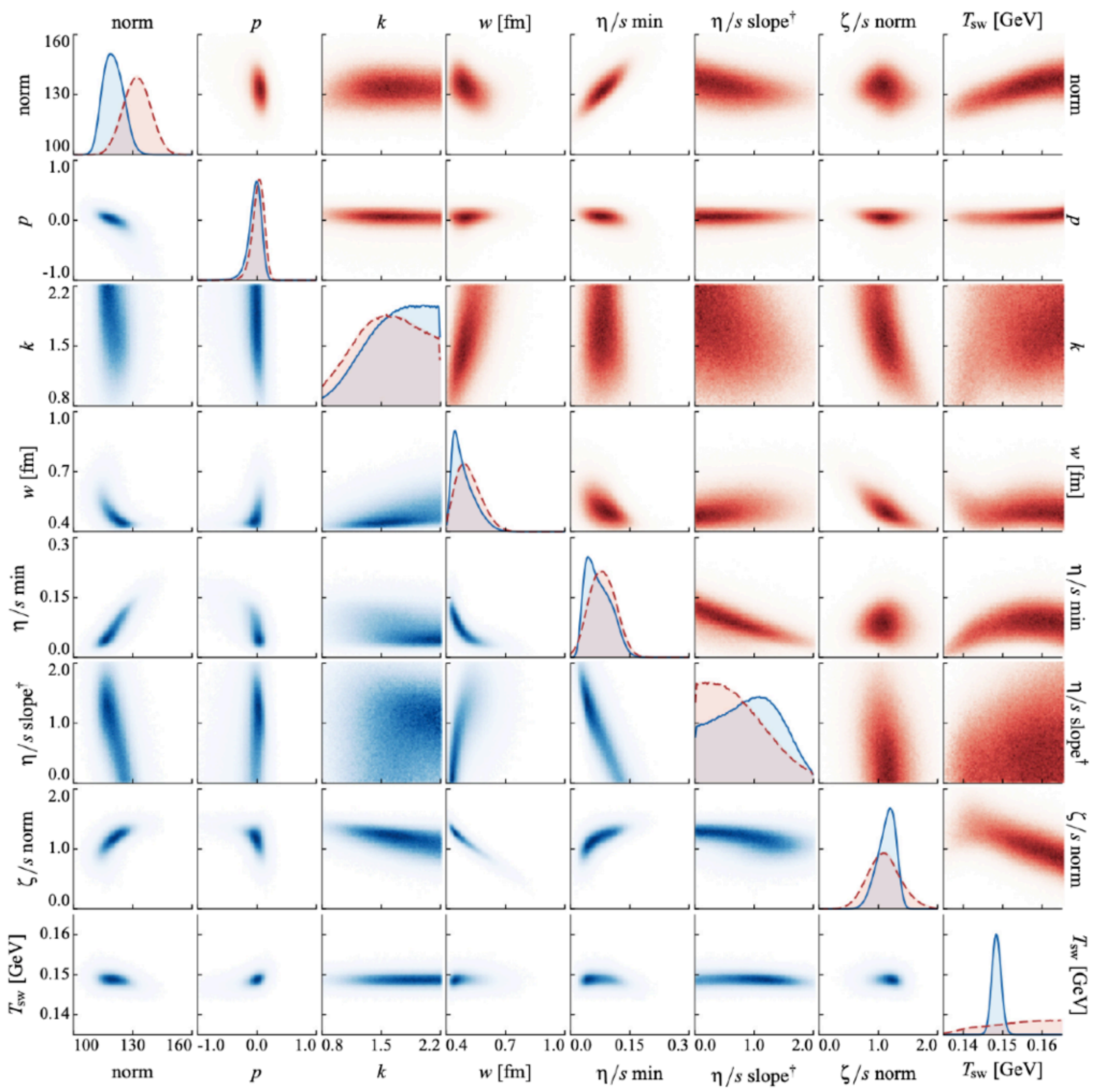
$$\hat{\theta} = \textbf{arg max}_{\theta}\{p(X \mid \theta)\}$$



Bayesian Inference

Inferring Dynamical Information

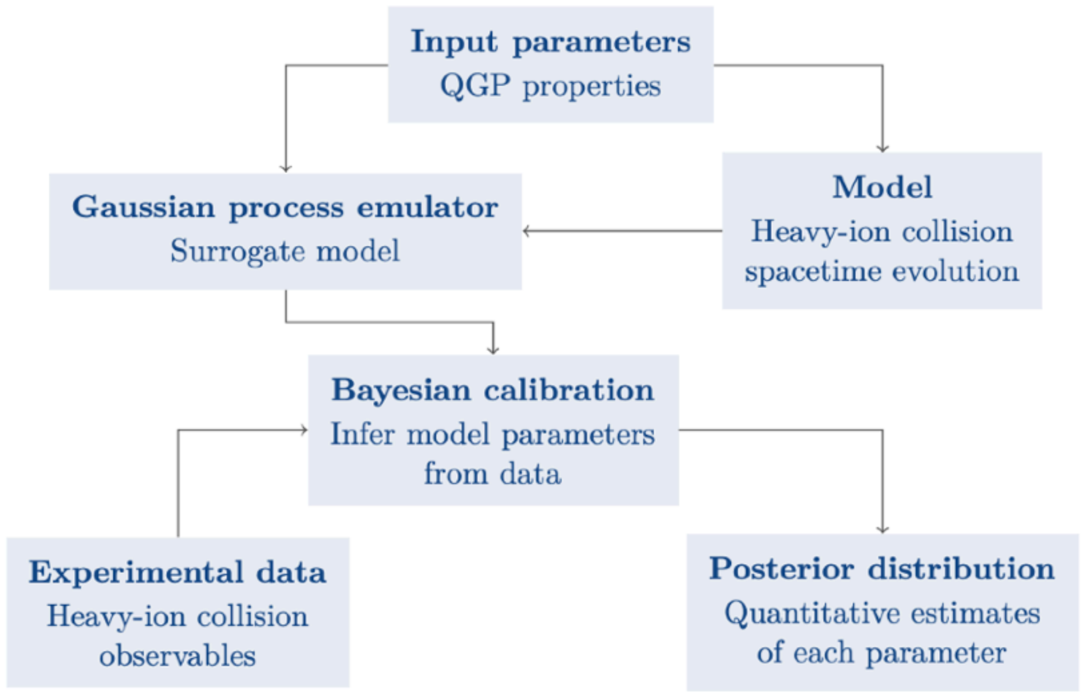




J. E. Bernhard, etc.(Duke Group), arXiv:1804.06469 and Nat Phy **15**, 1113 (2019).

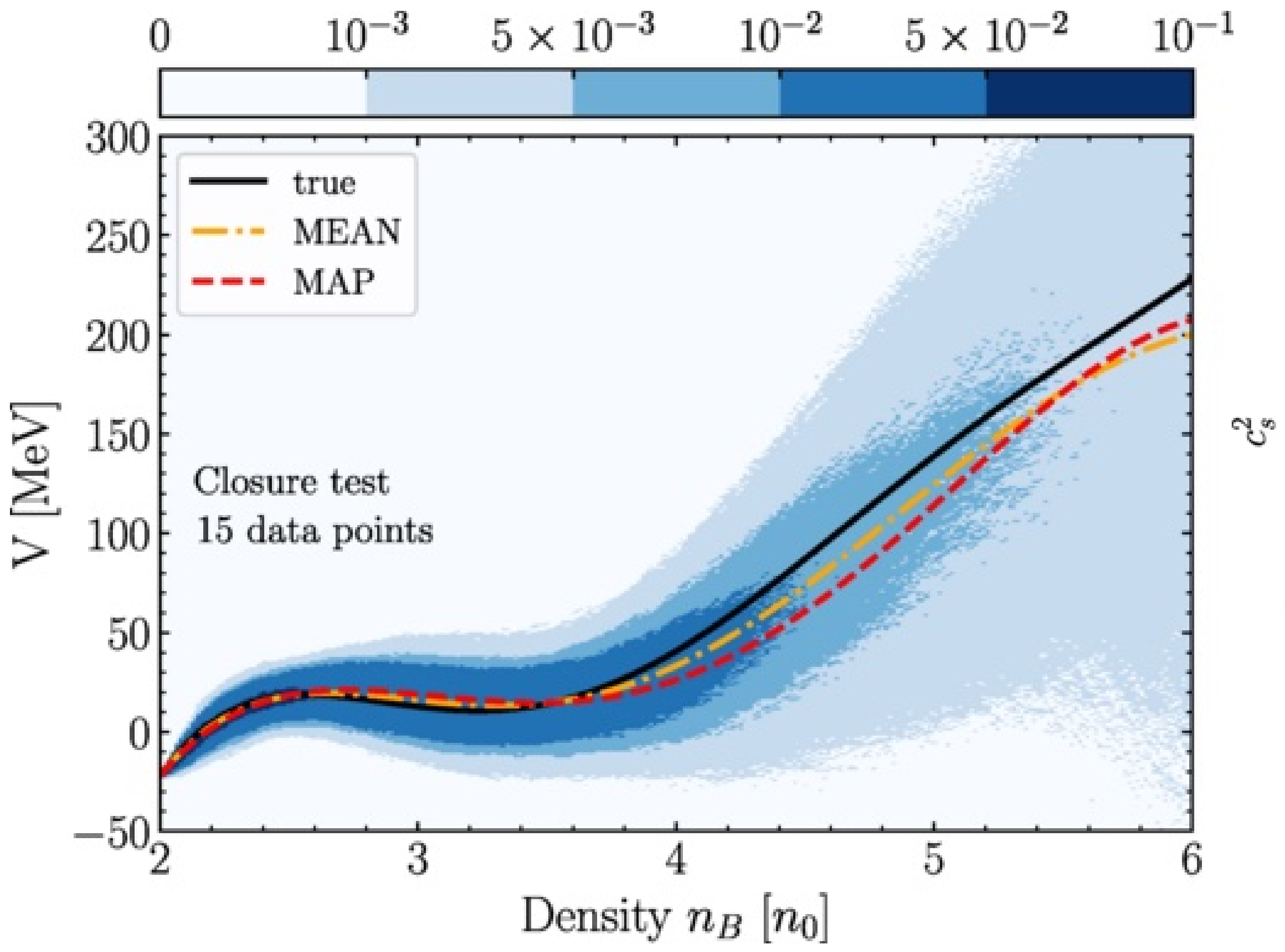
Bayesian Inference

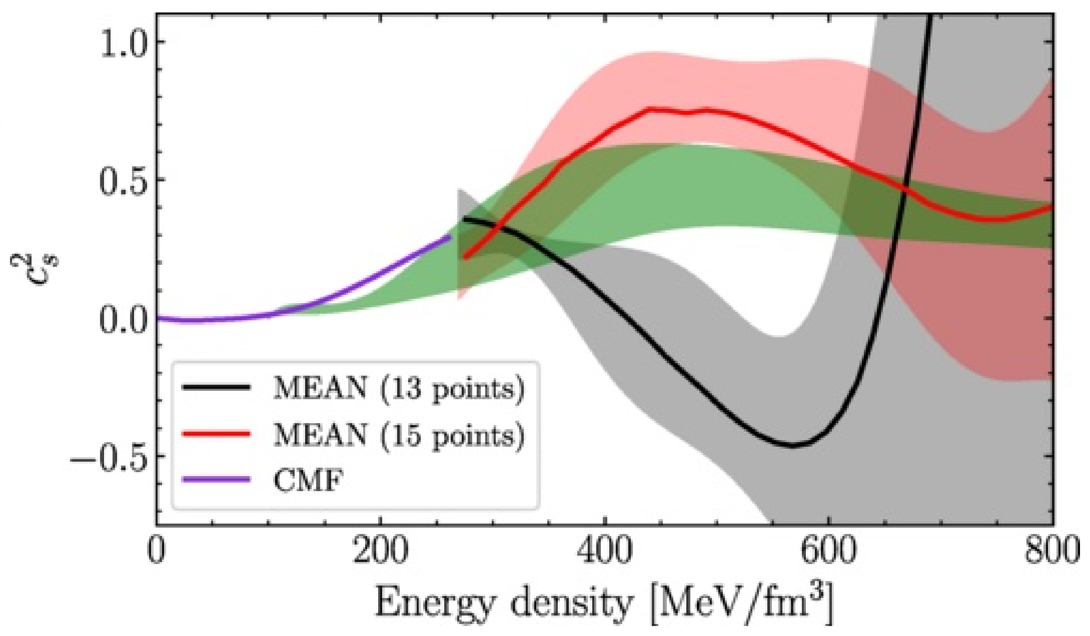
Inferring Dynamical Information



J. E. Bernhard, etc.(Duke Group), arXiv:1804.06469

Ectracting Dense Matter EoSs

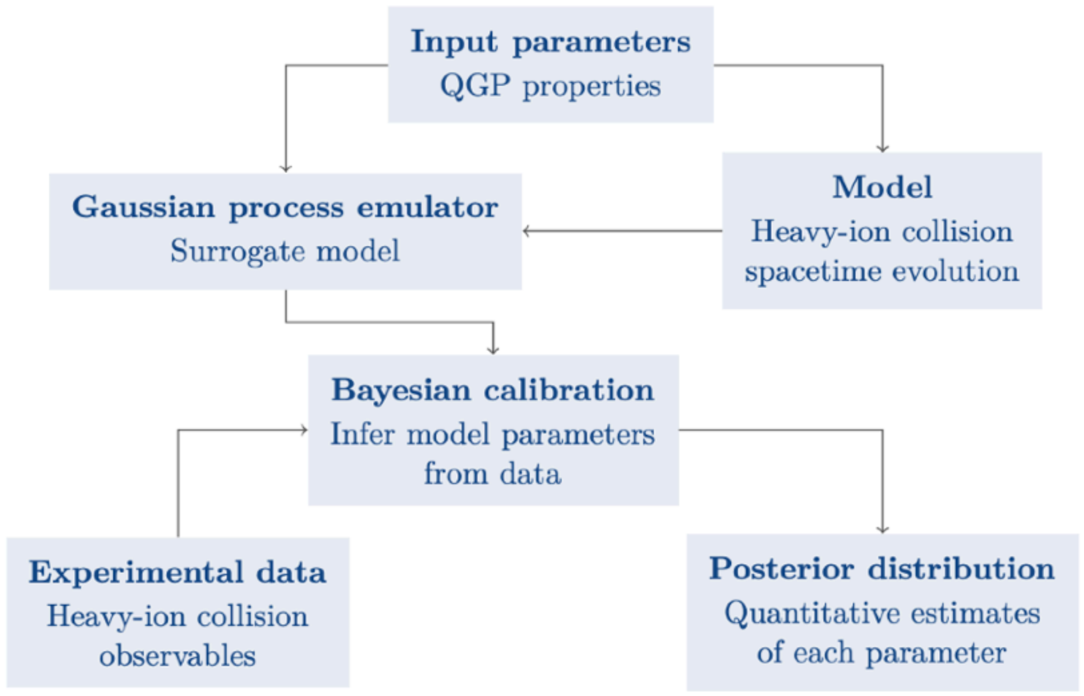




M. Omana Kuttan, J. Steinheimer, K. Zhou, and H. Stoecker, Phys. Rev. Lett. **131**, 202303 (2023).

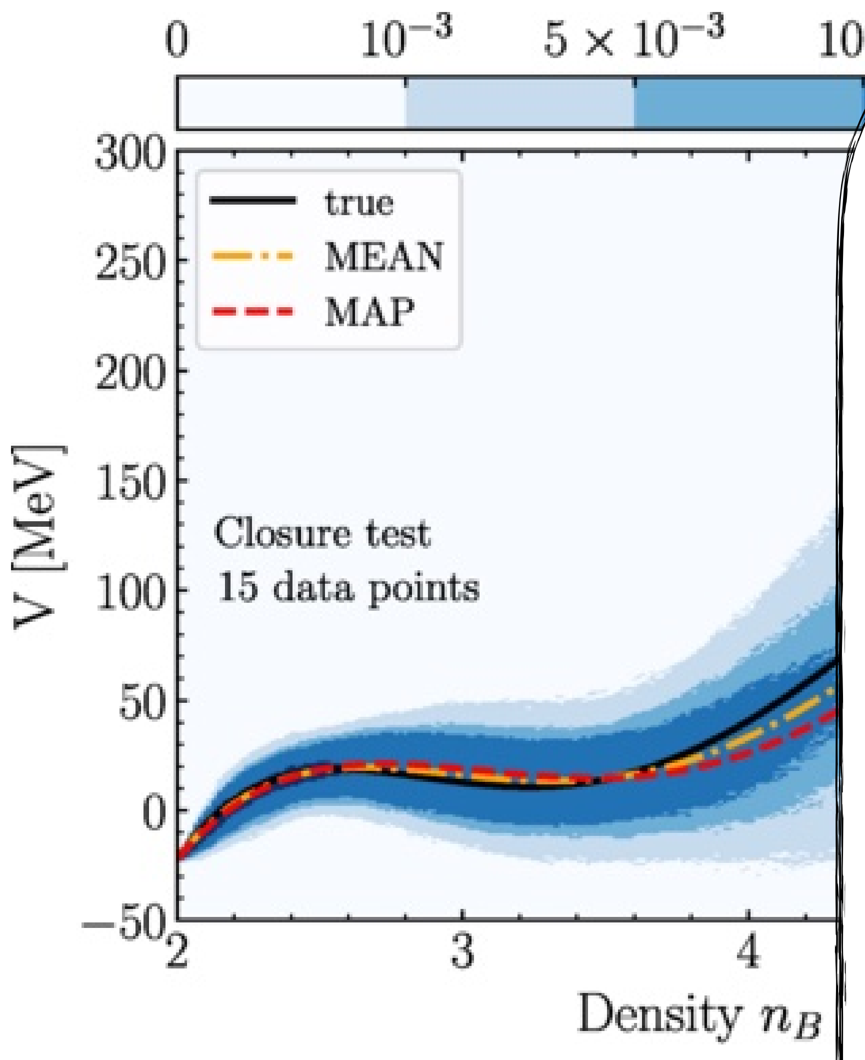
Bayesian Inference

Inferring Dynamical Information



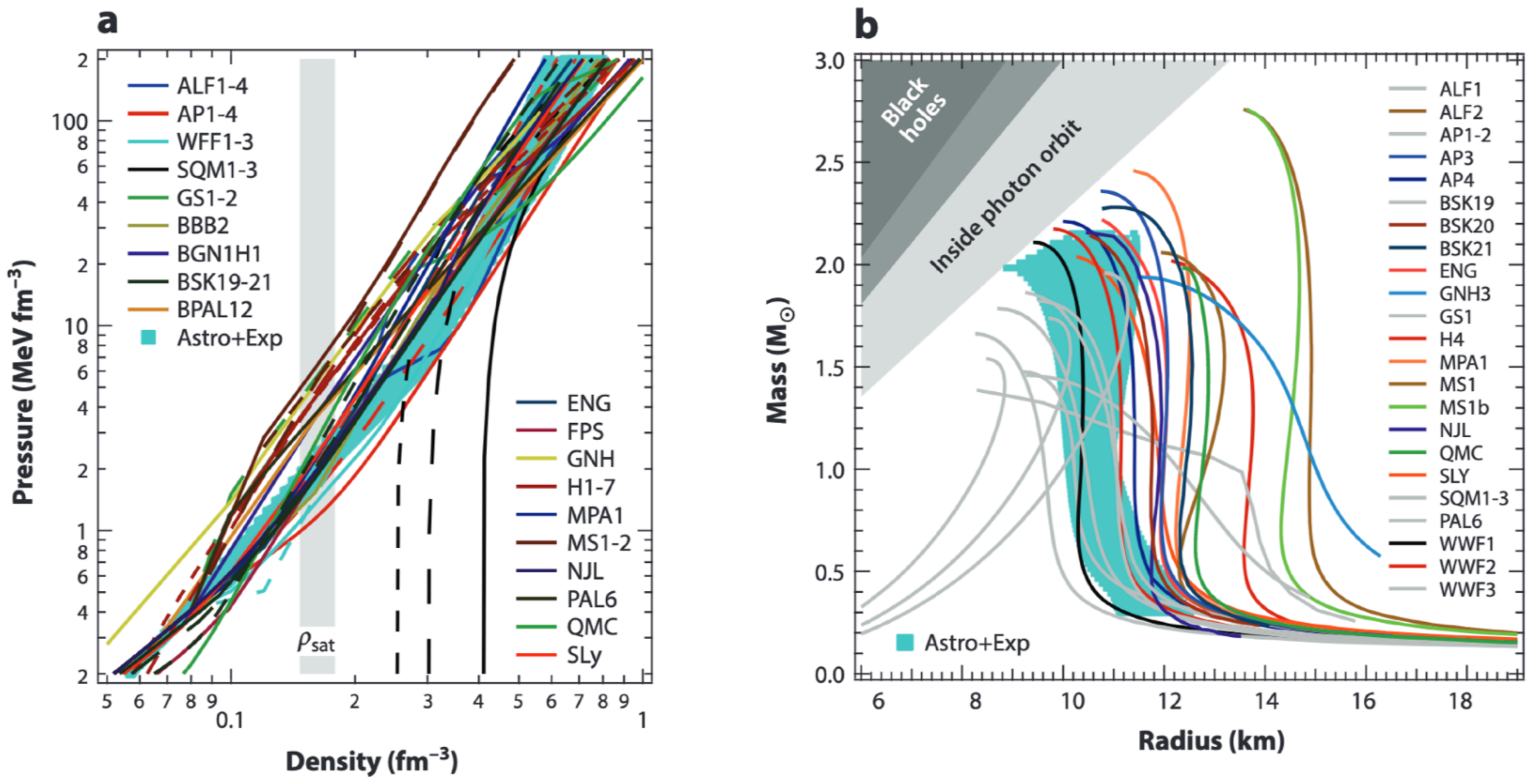
J. E. Bernhard, etc.(Duke Group), arXiv:1804.06469

Ectracting Dense Matter EoSs



M. Omana Kuttan, J. Ste

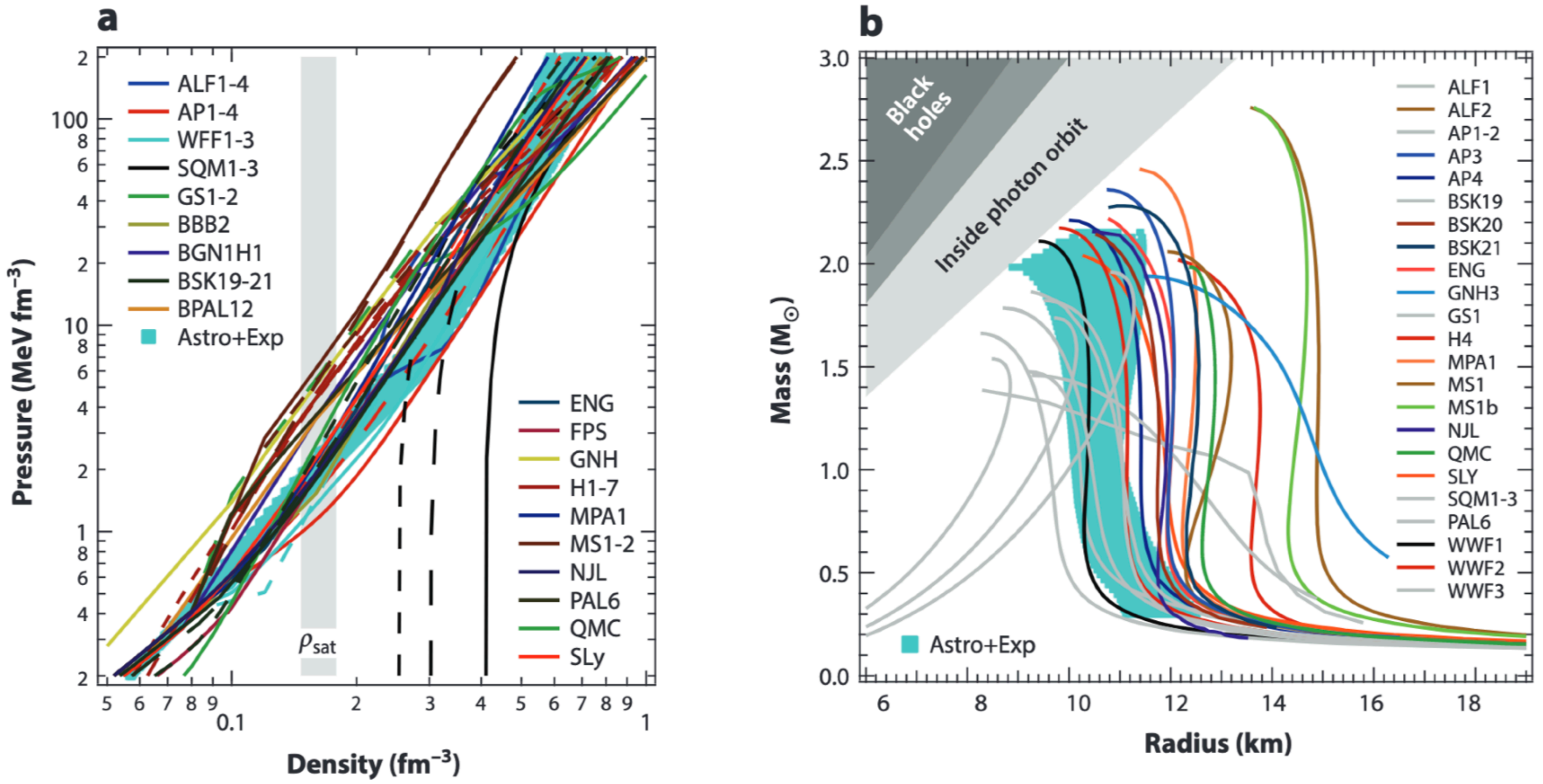
Building Nuclear Matter EoS



F. Özel and P. Freire, Annu. Rev. Astron. Astrophys. **54**, 401 (2016)

Bayesian Inference

Building Nuclear Matter EoS



F. Özel and P. Freire, Annu. Rev. Astron. Astrophys. **54**, 401 (2016)

Physics Parameters are Finite

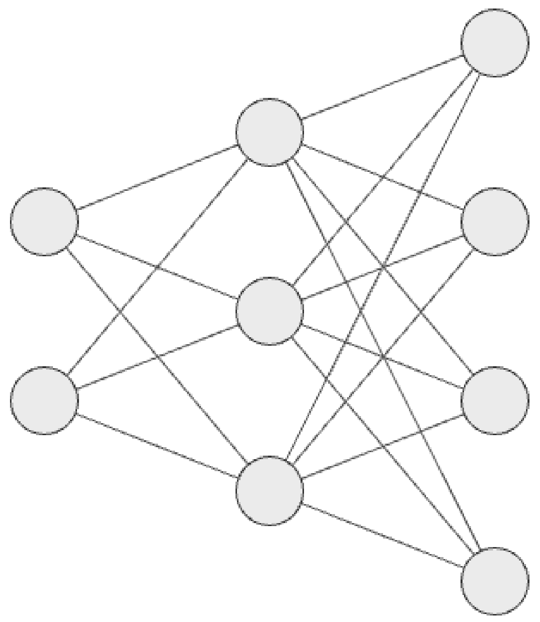
EoS, Wave-Function, Potential, …

Inference is Easy-To-Compute

ODEs, PDEs, Simulations, …

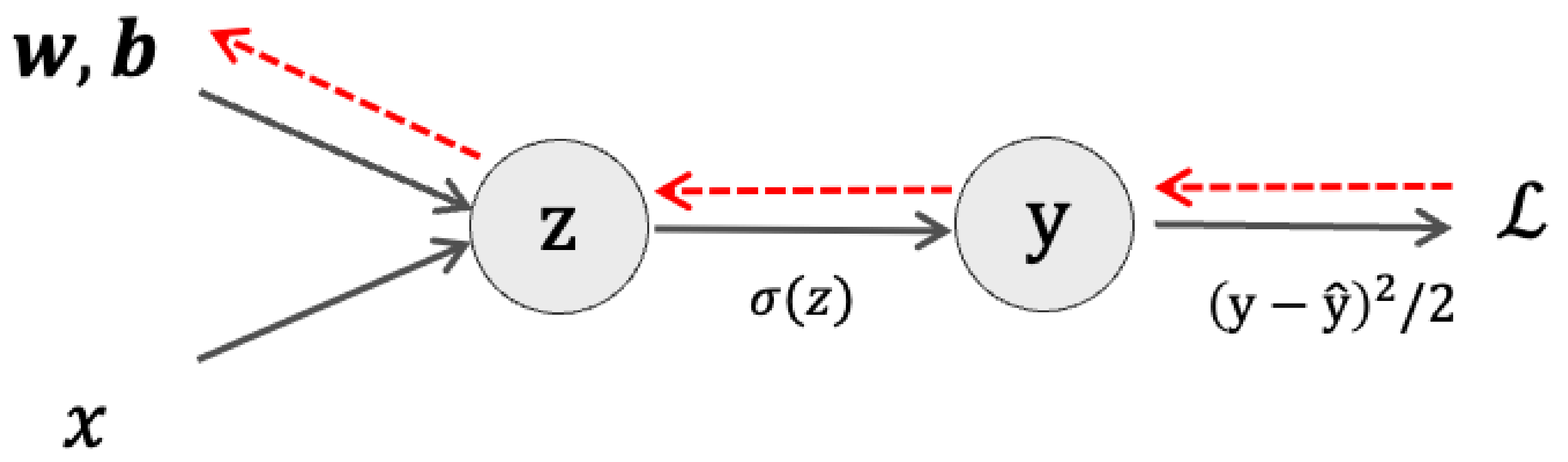
Physics-Driven Deep Learning

$$\hat{\theta} = \textbf{arg max}_{\theta}\{p(X \mid \theta)\}$$



f_θ Phys → $\tilde{O}$

← BP $\mathcal{L} = (O - \tilde{O})^2$



Deep Neural Network represented Physics, f_θ

Flexible Representation

Back-Propagation

Easy-To-Compute on GPUs

Physics-Driven Deep Learning

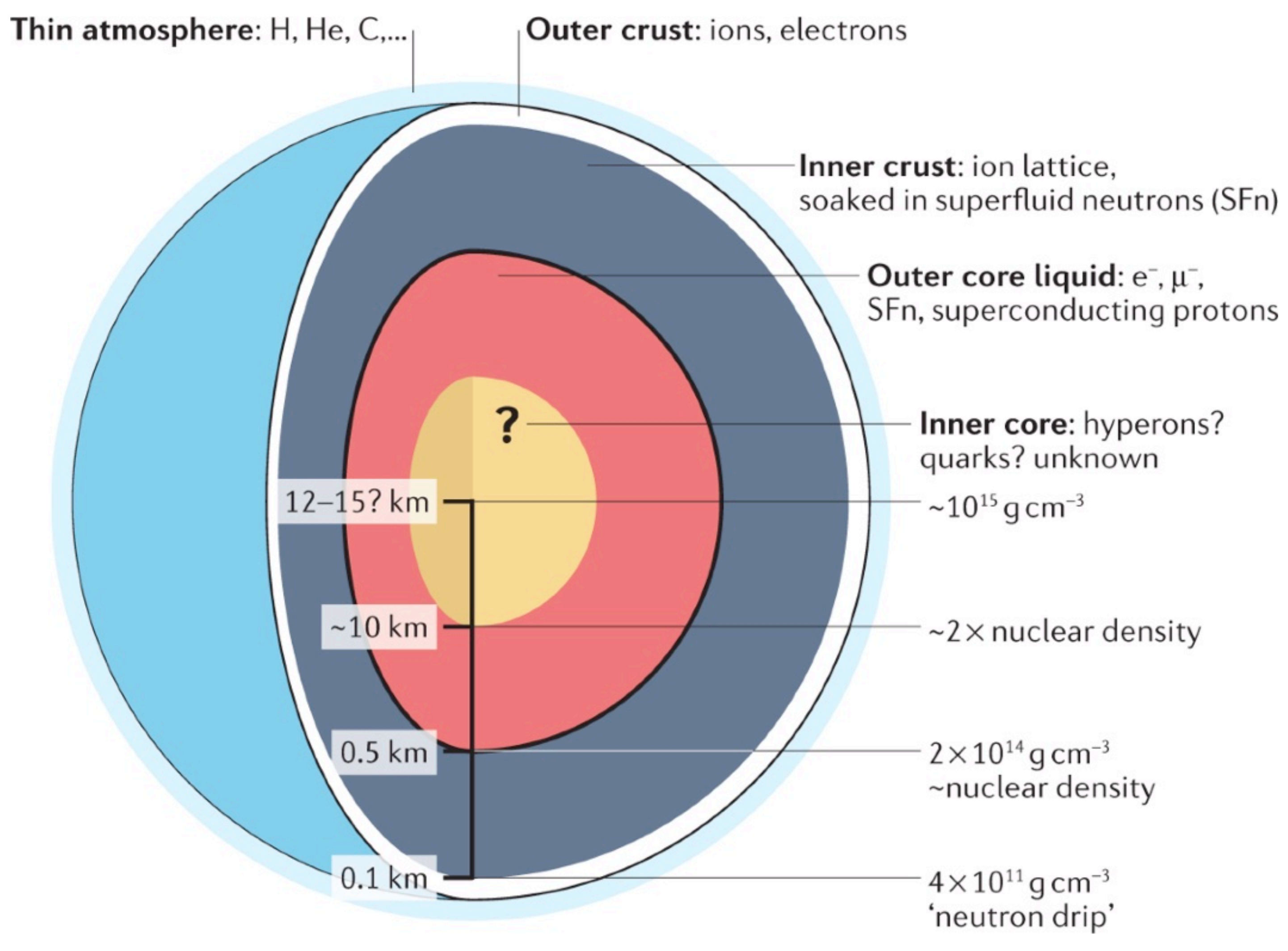
1. Building Neutron Star EoS

Tolman–Oppenheimer–Volkoff equations

$$\frac{dP}{dr} = -G\frac{m(r)\varepsilon(r)}{r^2}\left(1+\frac{P(r)}{\varepsilon(r)}\right)\left(1+\frac{4\pi r^3 P(r)}{m(r)}\right)\left(1-\frac{2Gm(r)}{r}\right)^{-1}$$

$$\frac{dm(r)}{dr} = 4\pi r^2 \epsilon(r)$$

Hydrostatic condition in each shell (dr)



Nat. Rev. Phys. 4, 237–246 (2022)

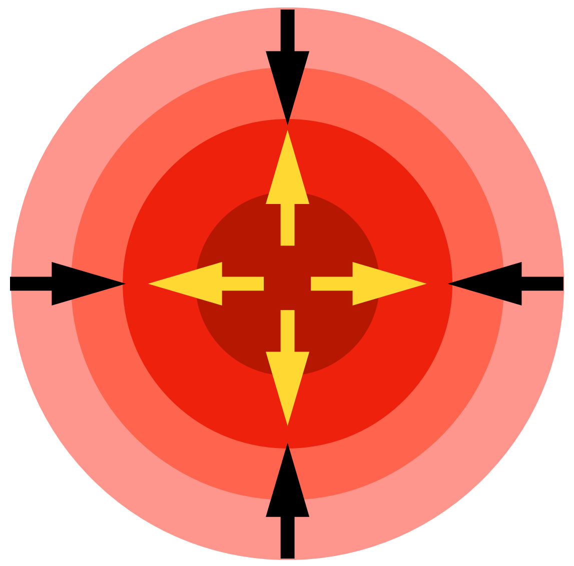
EoS $P(\varepsilon) = 0$

Core ↓ Surface

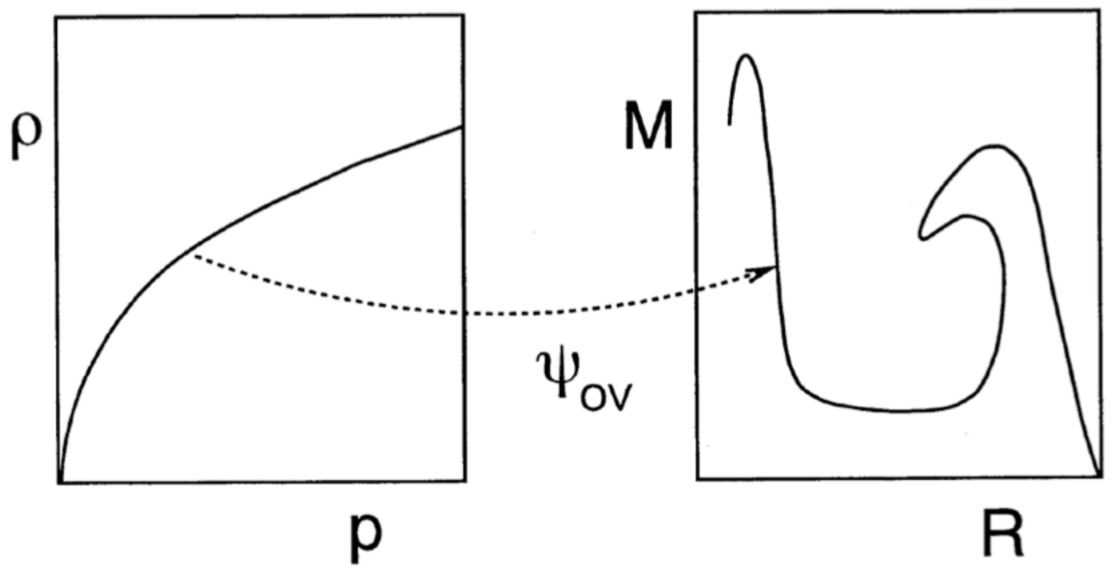
$r = 0, \varepsilon(r=0) = \varepsilon_c, P(r=0) = P(\varepsilon_c)$

$r = R, \varepsilon(r=R) \simeq 0, M = \int 4\pi r^2 \varepsilon(r) dr$

M, R



Pressure → ← Gravity

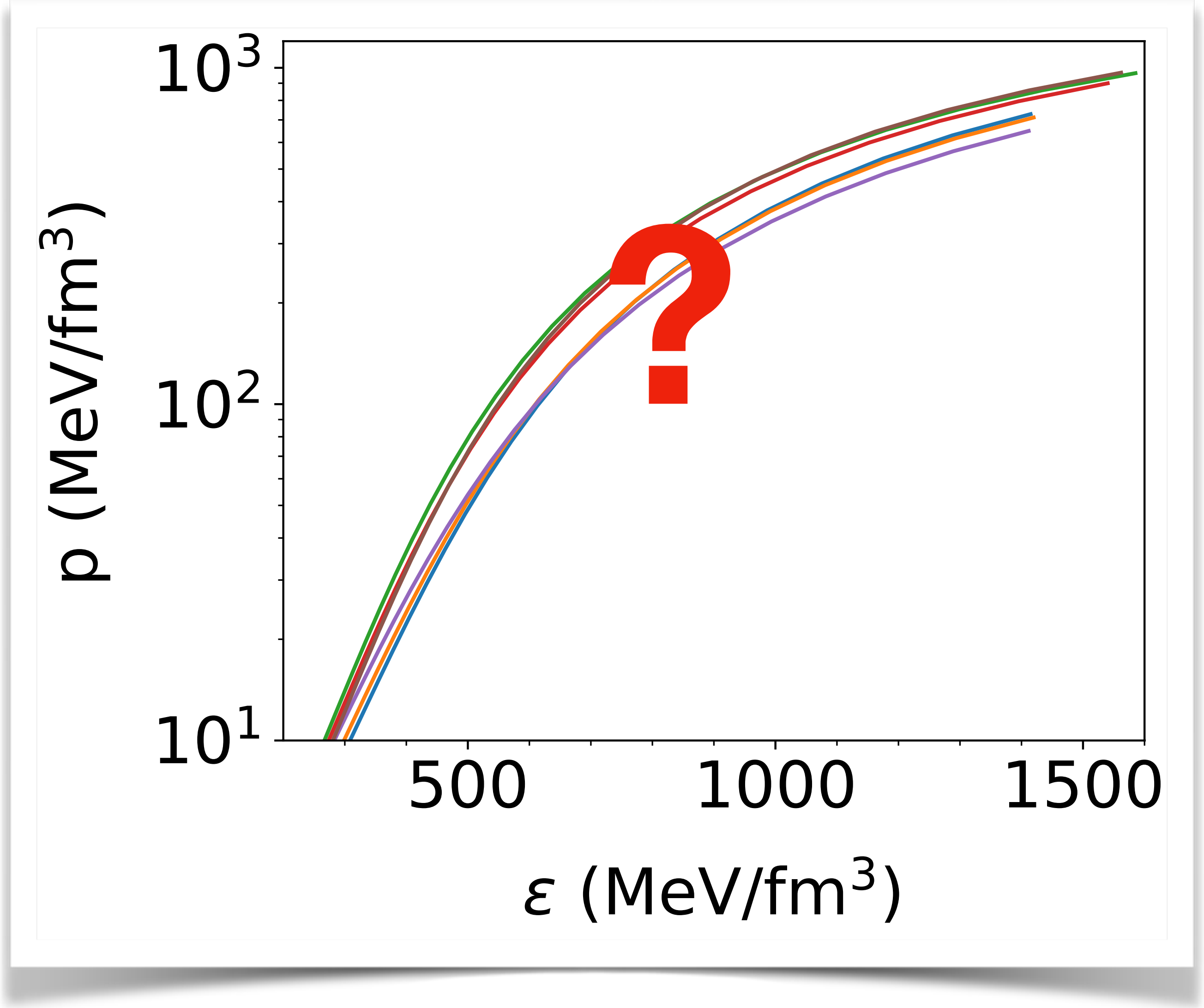


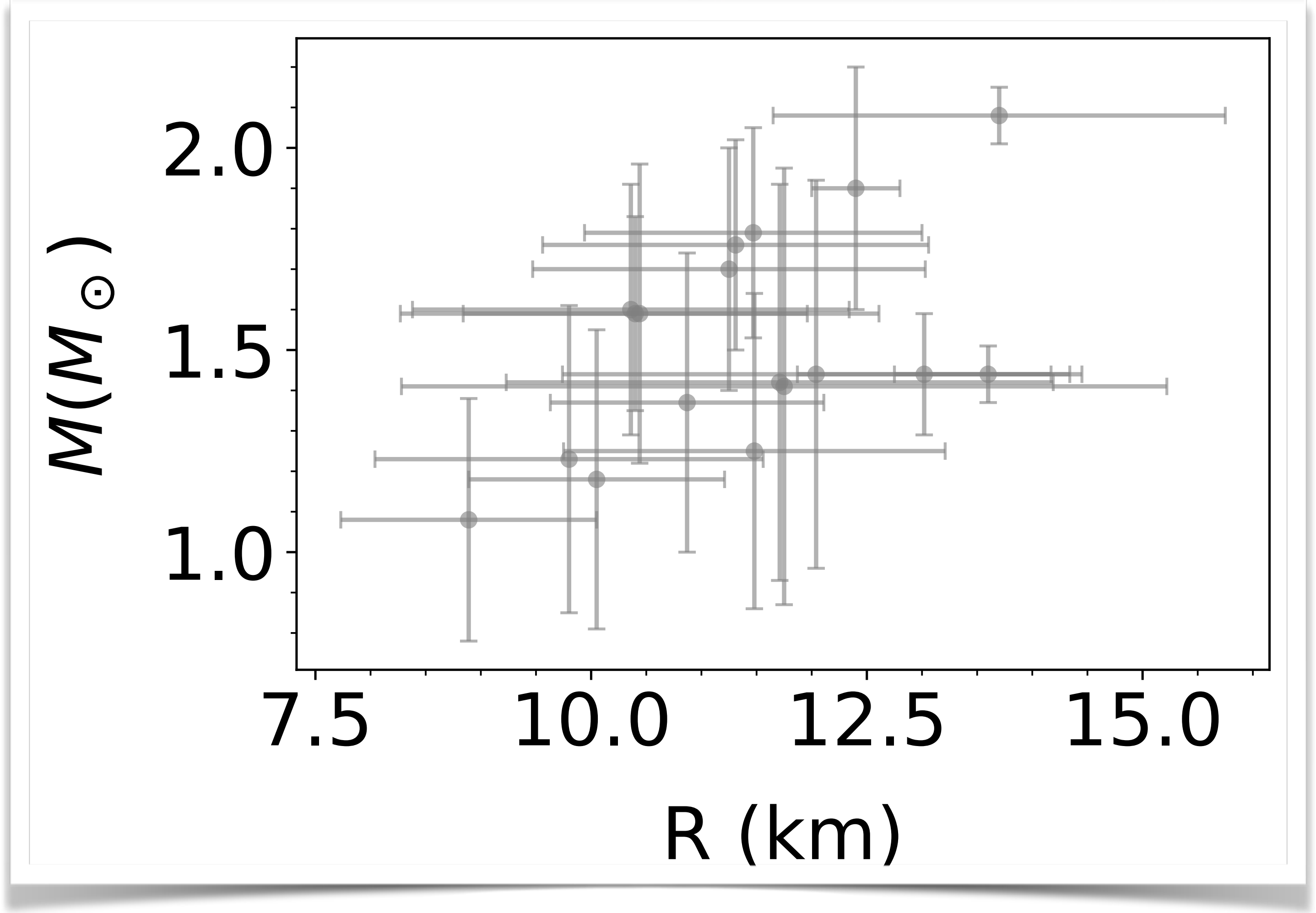
L. Lindblom, *A.J., 398, 569 (1992).*
If the **whole M(R)** is known, it's well-defined problem.

Physics-Driven Deep Learning

1. Building Neutron Star EoS

Tolman–Oppenheimer–Volkoff equations

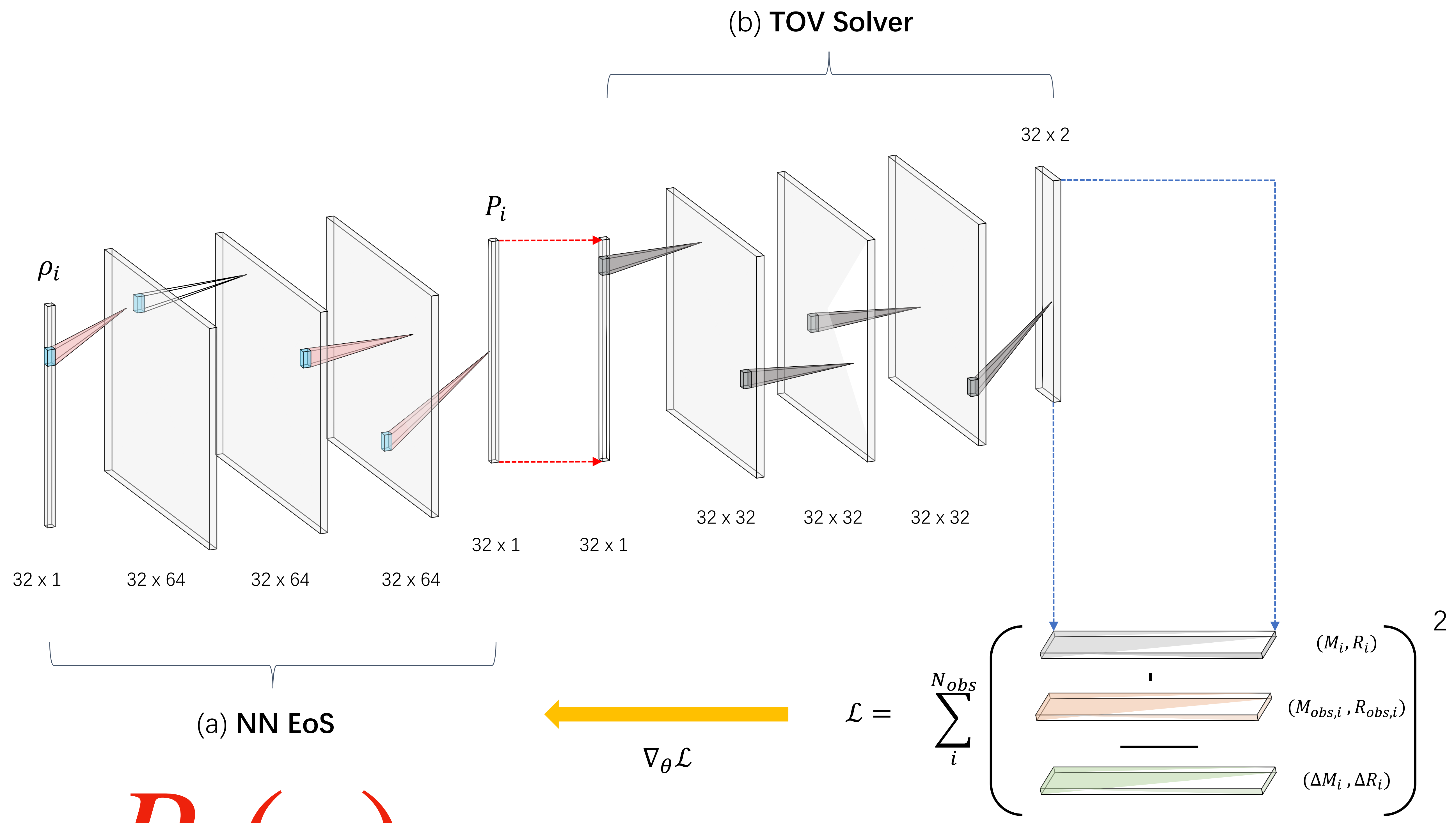




Physics-Driven Deep Learning

1. Building Neutron Star EoS

Phys. Rev. D 107, 083028; JCAP08 (2022) 071



$P_\theta(\rho)$

$$\mathscr{L} = \chi^2 = \sum_{i=1}^{N_{\text{obs}}} \frac{\left(M_i - M_{\text{obs},i}\right)^2}{\Delta M_i^2} + \frac{\left(R_i - R_{\text{obs},i}\right)^2}{\Delta R_i^2}$$

Loss function

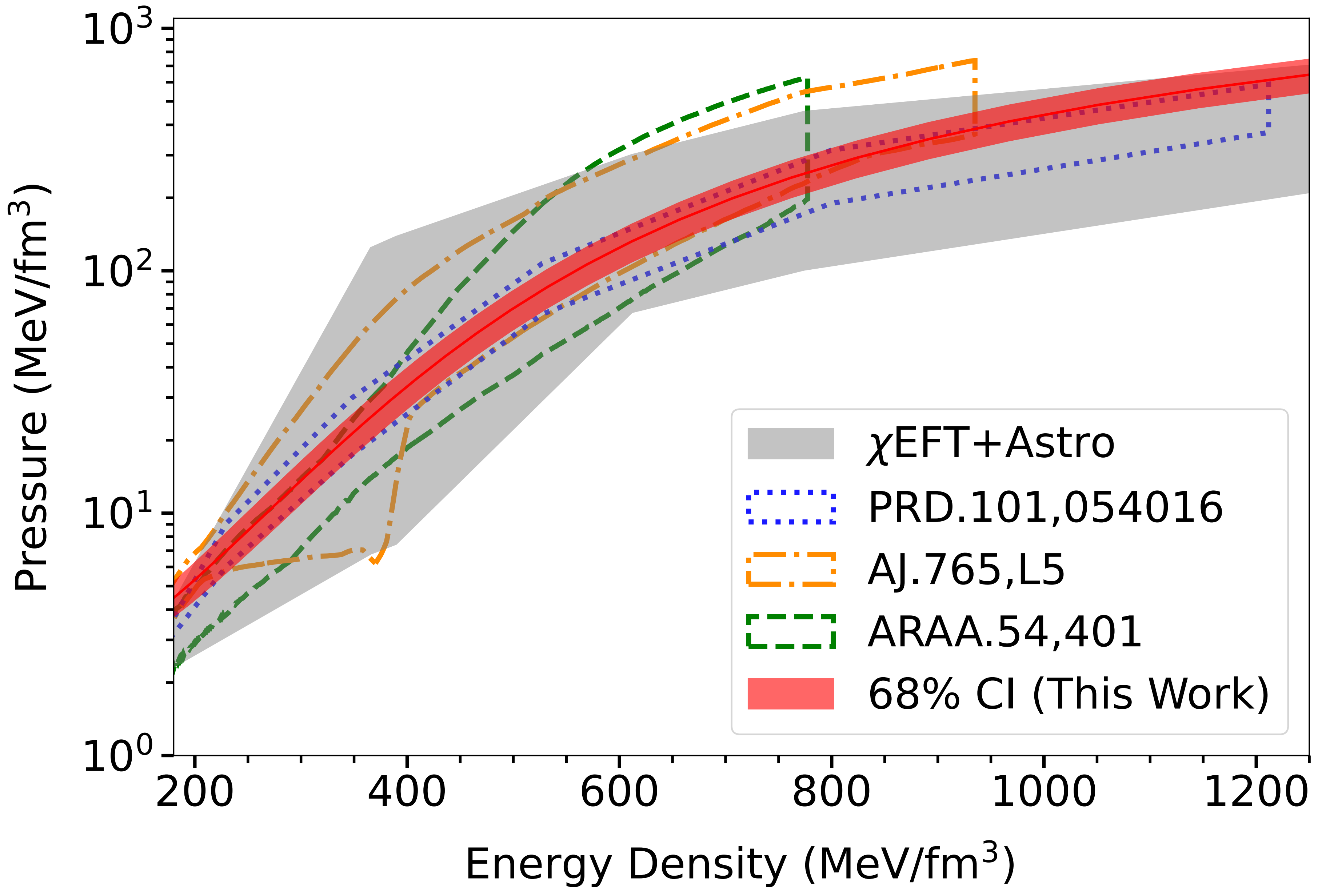
- (M_i, R_i) : predicted Mass-Radius
- $(M_{obs,i}, R_{obs,i})$: observed Mass-Radius
- $(\Delta M_i, \Delta R_i)$: uncertainties

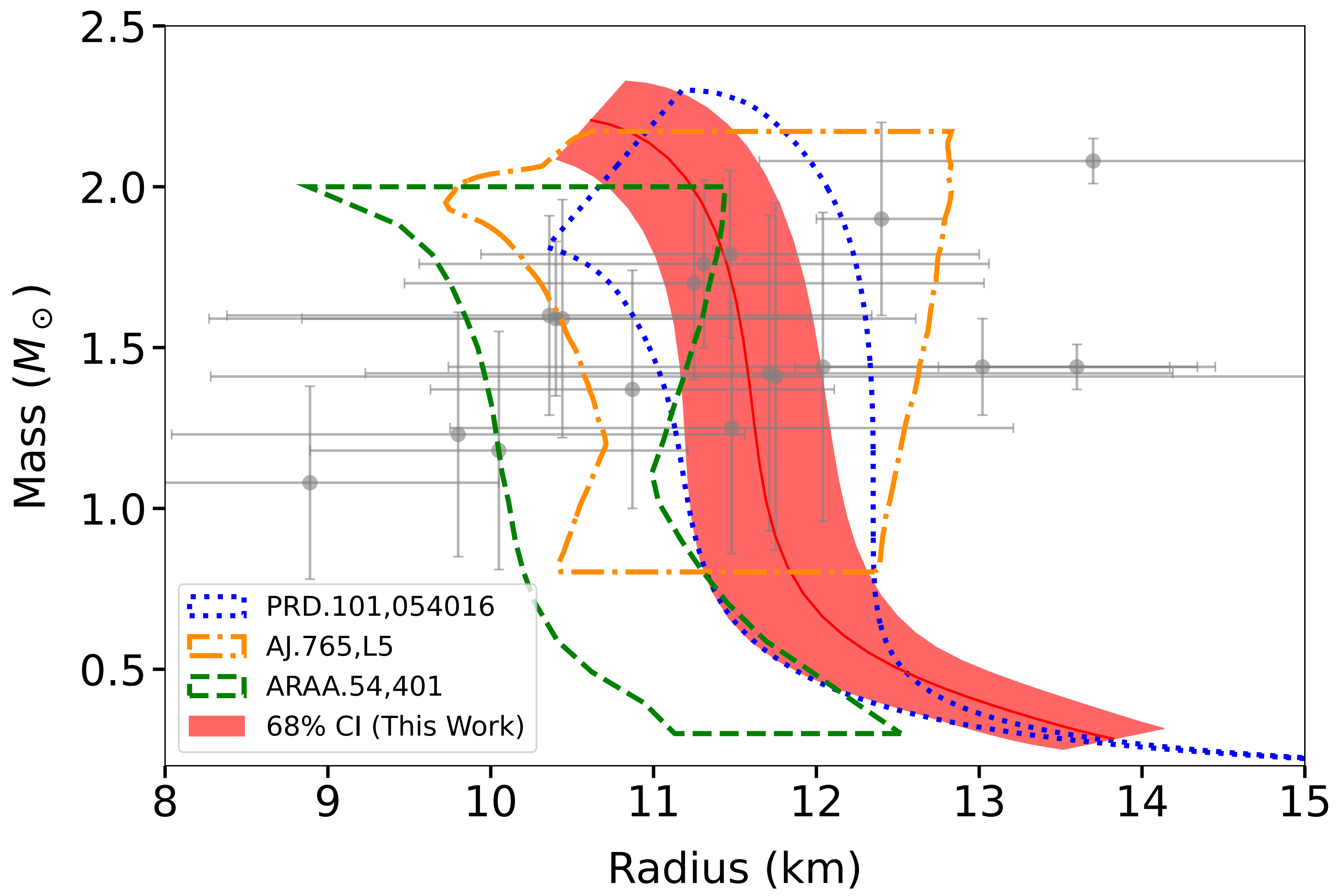
Physics-Driven Deep Learning

1. Building Neutron Star EoS

Phys. Rev. D 107, 083028

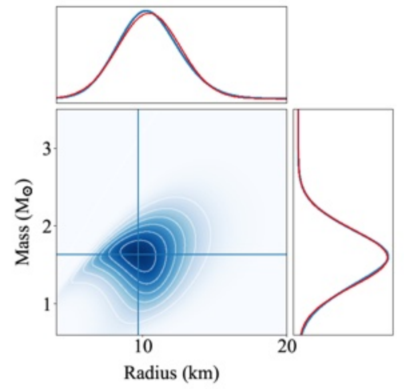
Blue dots: NN results, Fujimoto-Fukushima-Murase
Yellow and Green dashed lines: Bayesian Approaches





Our results: $R_{1.4} = 11.6 \pm 0.43$ **km (68% CI)**

18 (M_i, R_i) , sample size = **10k**
causality $(d\epsilon/dp < 1)$
Maximum mass $\geq 1.9\,M_\odot$



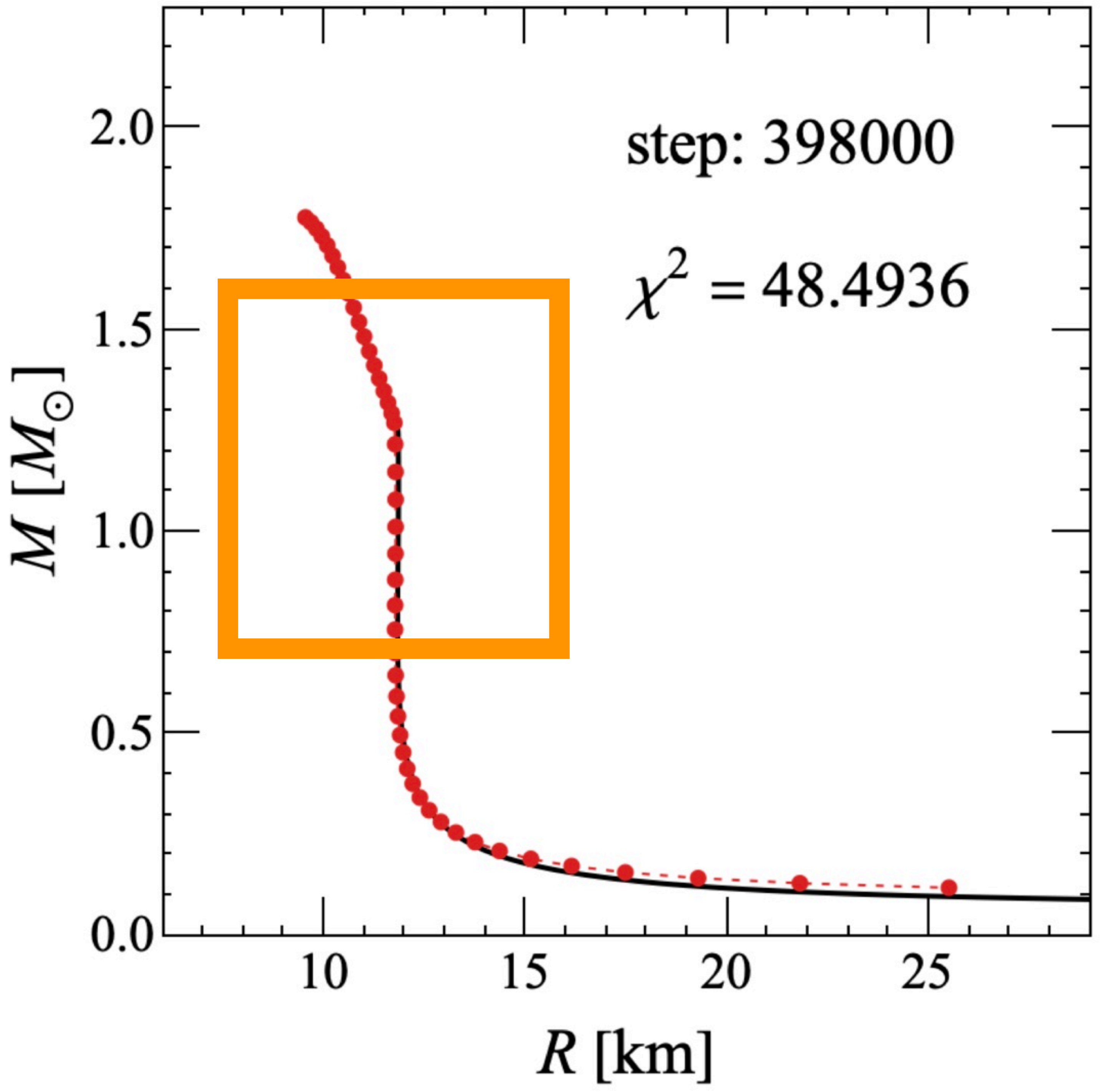
Observable	Mass($M_\odot$)	Radius(km)
M13	1.42±0.49	11.71±2.48
M28	1.08±0.30	8.89±1.16
M30	1.44±0.48	12.04±2.30
NGC 6304	1.41±0.54	11.75±3.47
NGC 6397	1.25±0.39	11.48±1.73
ωCen	1.23±0.38	9.80±1.76
4U 1608-52	1.60±0.31	10.36±1.98
4U 1724-207	1.79±0.26	11.47±1.53
4U 1820-30	1.76±0.26	11.31±1.75
EXO 1745-248	1.59±0.24	10.40±1.56
KS 1731-260	1.59±0.37	10.44±2.17
SAX J1748.9-2021	1.70±0.30	11.25±1.78
X5	1.18±0.37	10.05±1.16
X7	1.37±0.37	10.87±1.24
4U 1702-429	1.90±0.30	12.40±0.40
PSR J0437–4715	1.44±0.07	13.60±0.85
PSR J0030+0451	1.44±0.15	13.02±1.15
PSR J0740+6620	2.08±0.07	13.70±2.05

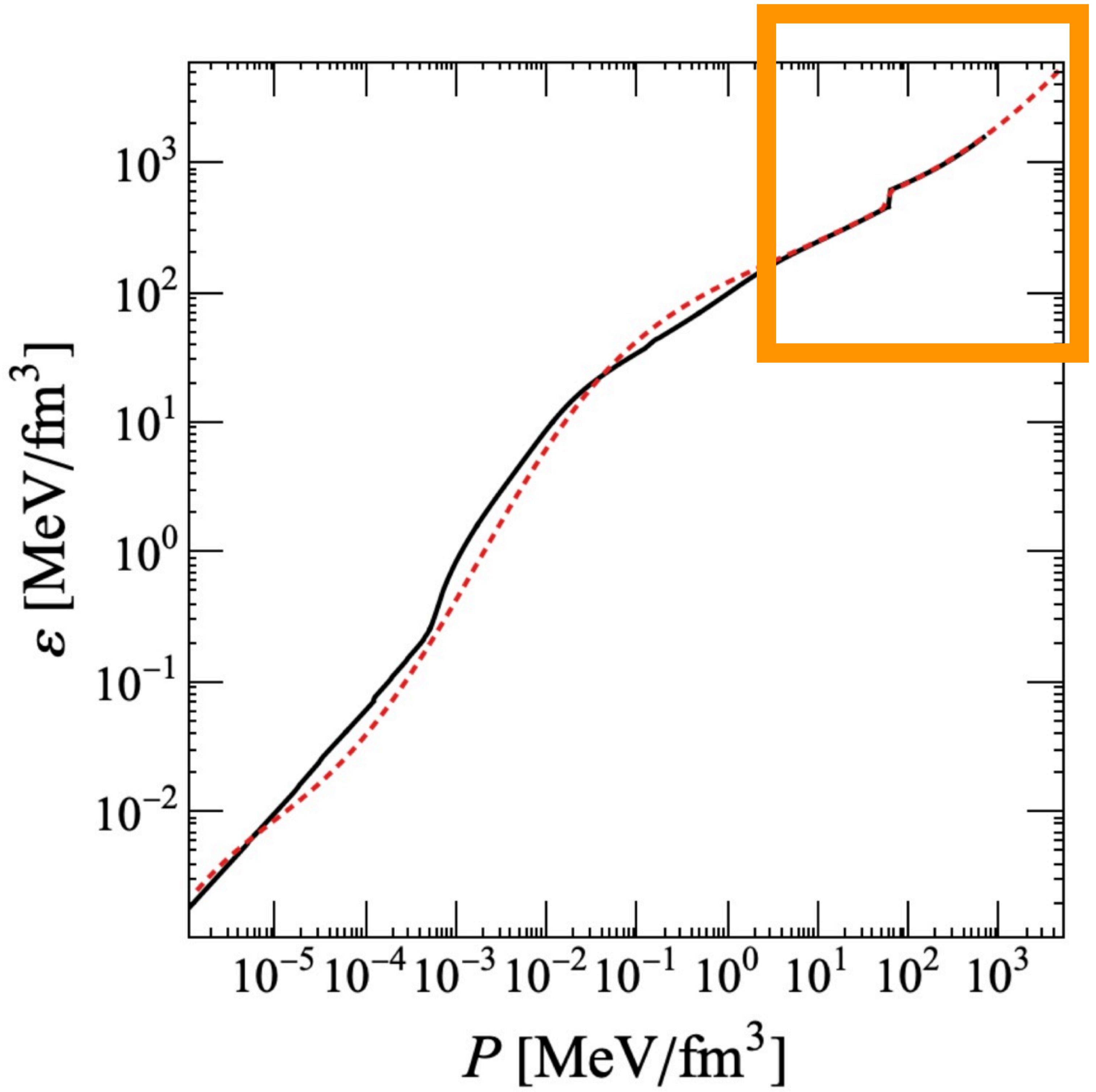
Physics-Driven Deep Learning

1. Building Neutron Star EoS

in Preparation

with Shuzhe Shi, Zidu Lin, etc.





Able to capture first-order phase transition !

48 Neutron Stars

(24 in $M > M_\odot$)

Represent Speed of Sound

1.Microscopicly stable condition, $\frac{dp}{d\epsilon} \geq 0$

2.Causality condition, $\frac{dp}{d\epsilon} = \frac{c_s^2}{c^2} < 1$

Representing $c_s(\varepsilon) = \sigma(y(\varepsilon))$ with neural netowrks$y(\varepsilon)$, above two conditions can be naturally met with the ***sigmoid*** activation function, $\sigma(x) = 1/(1+e^{-x})$.

Physics-Driven Deep Learning

2. Reconstructing Spectral Function

Correlation Function

$$\bar{D}_{\eta\eta'} = \mathrm{Tr}(\mathrm{T}_\tau[J_\eta(\tau,\mathrm{x})J^\dagger_{\eta'}(0,0)]e^{-H/T})/Z$$

$$\equiv T\sum_n \int \frac{d^3k}{(2\pi)^3} D_{\eta\eta'}(i\omega_n, \mathrm{k})e^{-i(\omega_n\tau - \mathrm{k}\cdot\mathrm{x})}$$

M. Asakawa, Y. Nakahara, and T. Hatsuda, Prog. Part. Nucl. Phys. 46, 459 (2001)

$\mathbf{k} = 0, \eta = \eta'$ → **Spectrum representation**

$\mathbf{k} = 0, \eta = \eta', T \to 0$ → **Energy eigenspace**

Spectrum representation

$$G(\tau, T) = \int_0^\infty K(\omega,\tau,T)\rho(\omega,T)d\omega$$

$$K(\omega,\tau,T) = \frac{\cosh\omega(\tau - \frac{1}{2T})}{\sinh\frac{\omega}{2T}}$$

Thermal Kernel

$T \to 0$

Energy eigenspace

$$C(\tau) = \sum_{n=0} e^{-E_n\tau} | < \Omega_0 | J(0) | \Omega_n > |^2$$

$$\equiv \sum_{n=0} |c_n|^2 e^{-E_n\tau}$$

Discretization ↑

$$G(\tau) = \int_0^\infty e^{-\tau\omega}\rho(\omega)d\omega$$

Laplace Kernel

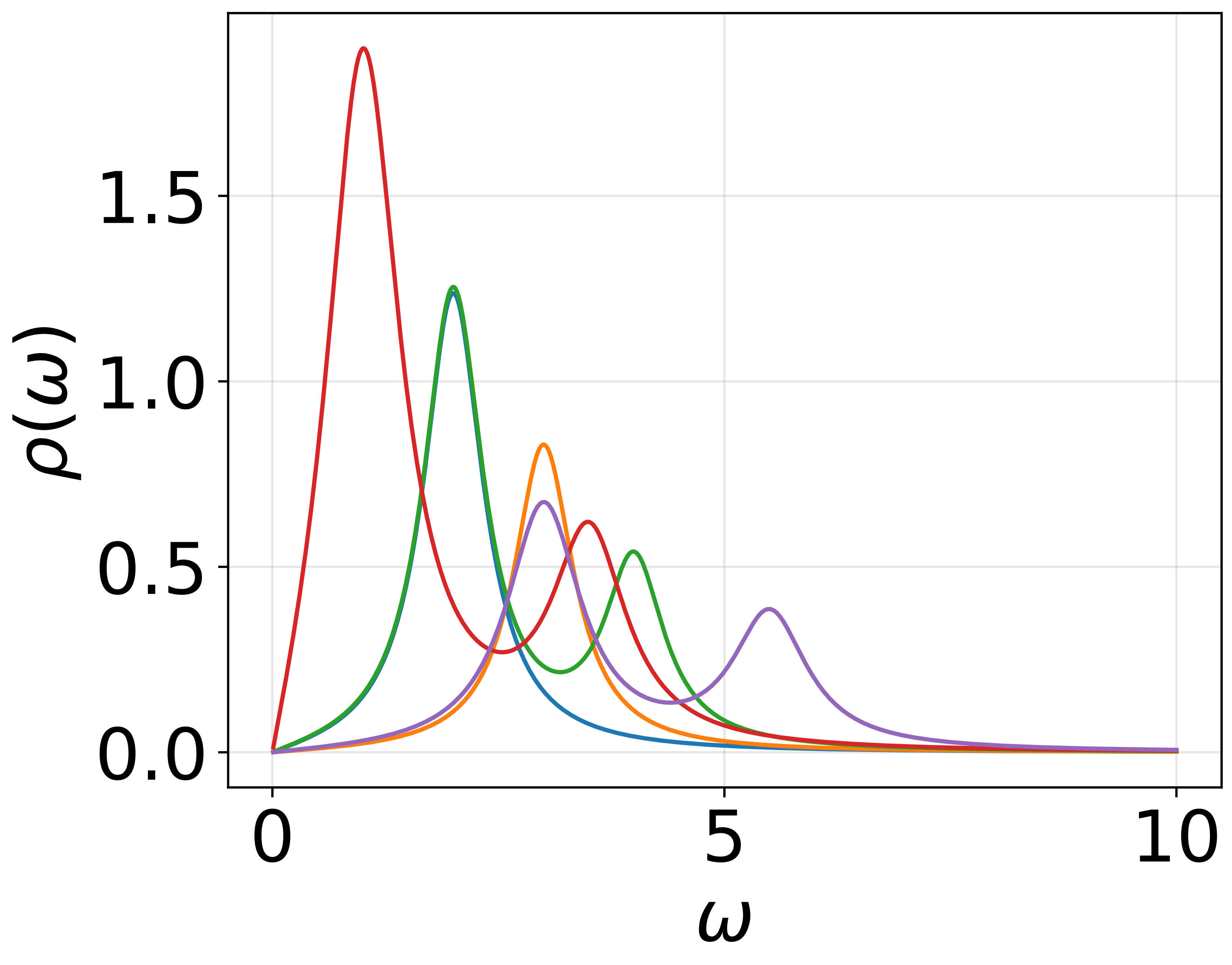
Spectral Function $\rho(\omega),\quad E_n$

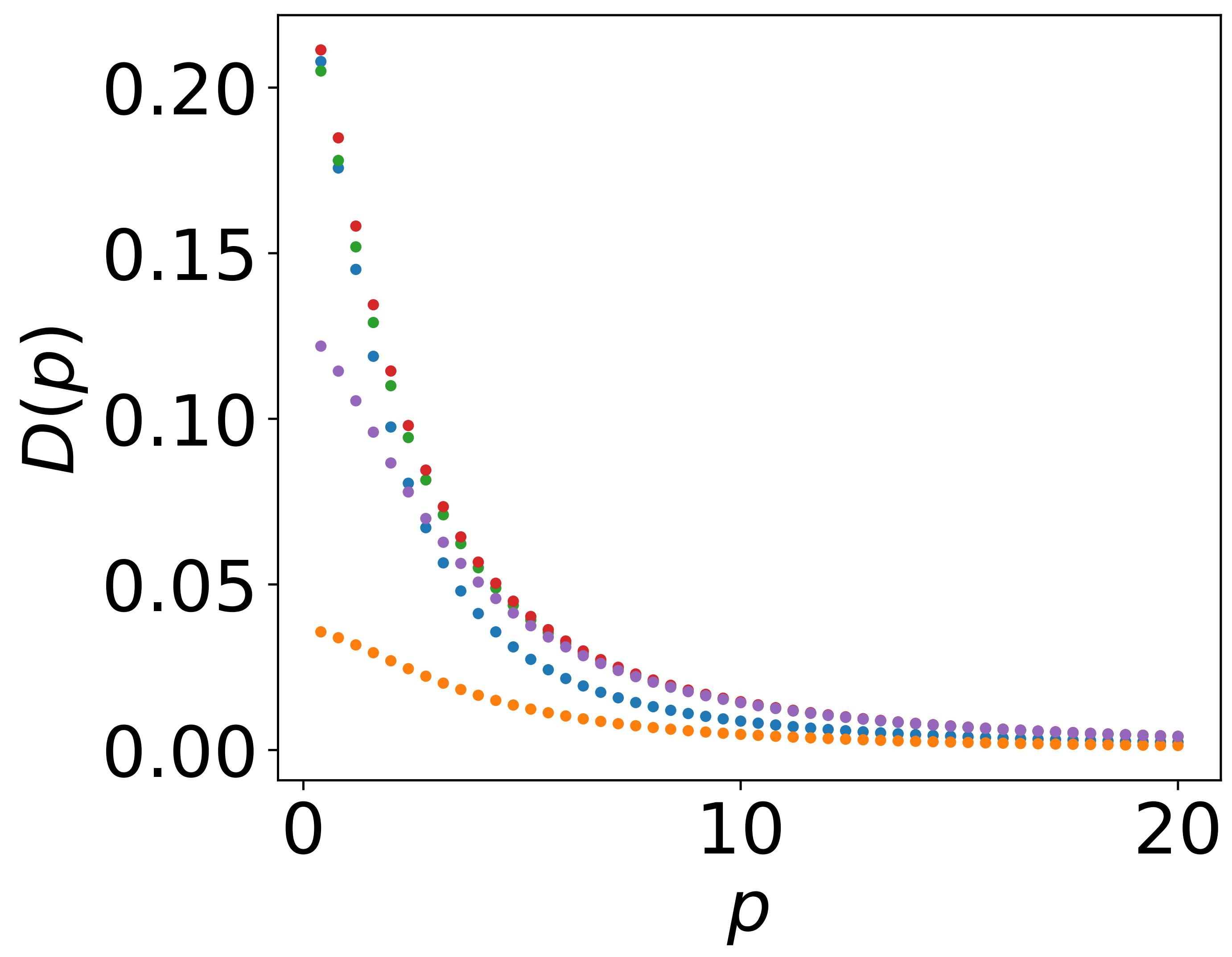
Physics-Driven Deep Learning

2. Reconstructing Spectral Function

Kallen—Lehmann(KL) **representation**

$$D(p) \equiv \int_0^\infty K(p,\omega)\rho(\omega)\mathrm{d}\omega \quad K(p,\omega) = \frac{\omega}{\pi(\omega^2+p^2)}$$



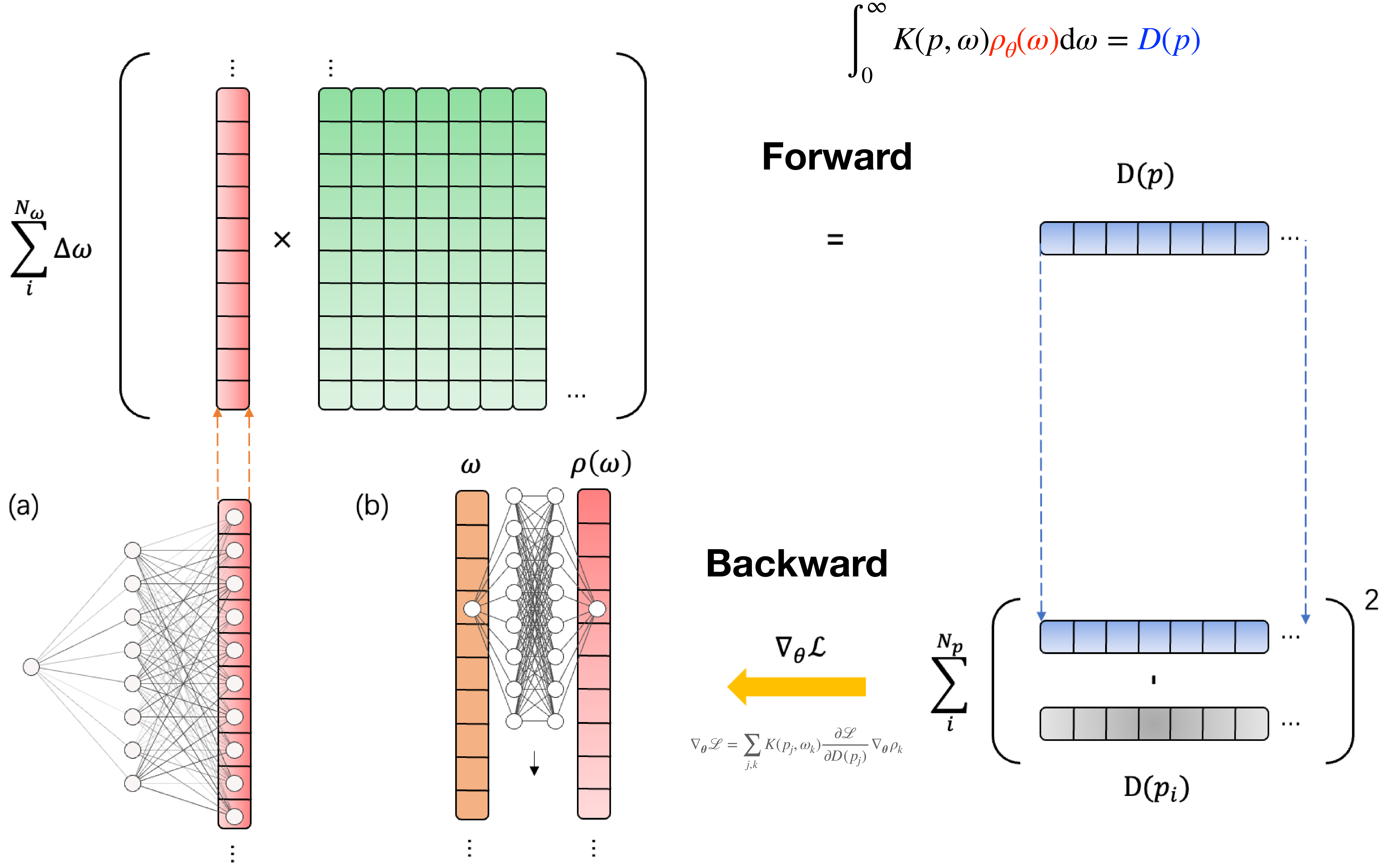


Physics-Driven Deep Learning

2. Reconstructing Spectral Function

Phys. Rev. D 106, L051502

$$\int_0^\infty K(p,\omega)\rho_\theta(\omega)\mathrm{d}\omega = D(p)$$



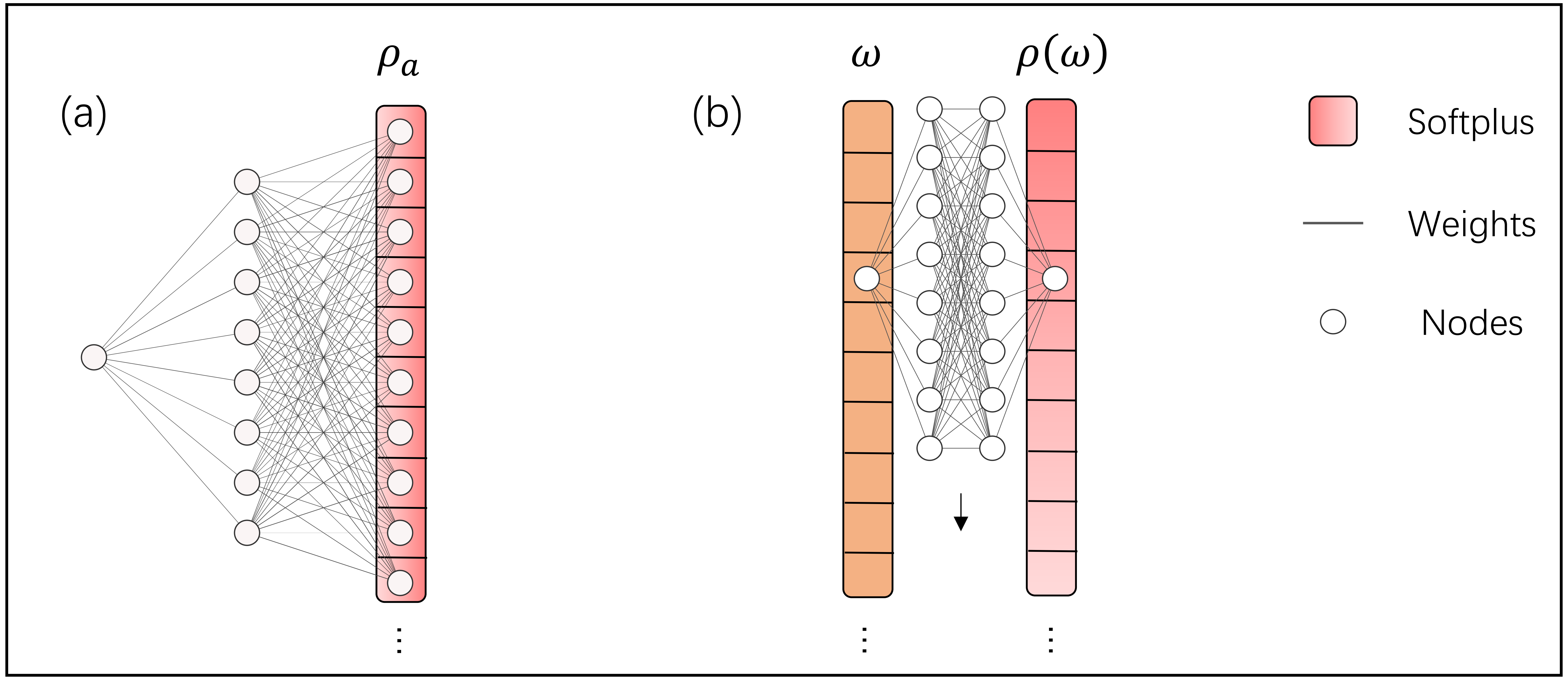
χ^2 Loss function

$$\mathscr{L} = \sum_i^{N_p} w_i(\mathrm{D}_i - D(p_i))^2$$

Physics-Driven Deep Learning

2. Reconstructing Spectral Function

Phys. Rev. D 106, L051502



NN : $(\rho_1, \rho_2, \cdots, \rho_{N_\omega})$

Differentiable variables : Network weights $\{\theta\}$

Adam, L2 ($\lambda = 10^{-3} \rightarrow 10^{-8}$), Smoothness ($\lambda_s = 10^{-4} \rightarrow 0$)

width = 64 and depth = 3 with bias

NN-P2P : $\rho(\omega)$

Differentiable variables : Network weights $\{\theta\}$

Adam, L2 ($\lambda = 10^{-6} \rightarrow 0$)

width = 64 and depth = 3 with bias

Regularization

L2 : $\lambda \mid \theta \mid_2^2$

Smoothness: $\lambda_s \sum_i^{N_\omega} (\rho_i - \rho_{i-1})^2$

Gradient-based Optimization

Adam : $\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t$

Physical Prior

Positive-defined condition(for hadrons)

Softplus $log(1 + e^x)$

Continuity

NN-P2P

Physics-Driven Deep Learning

2. Reconstructing Spectral Function

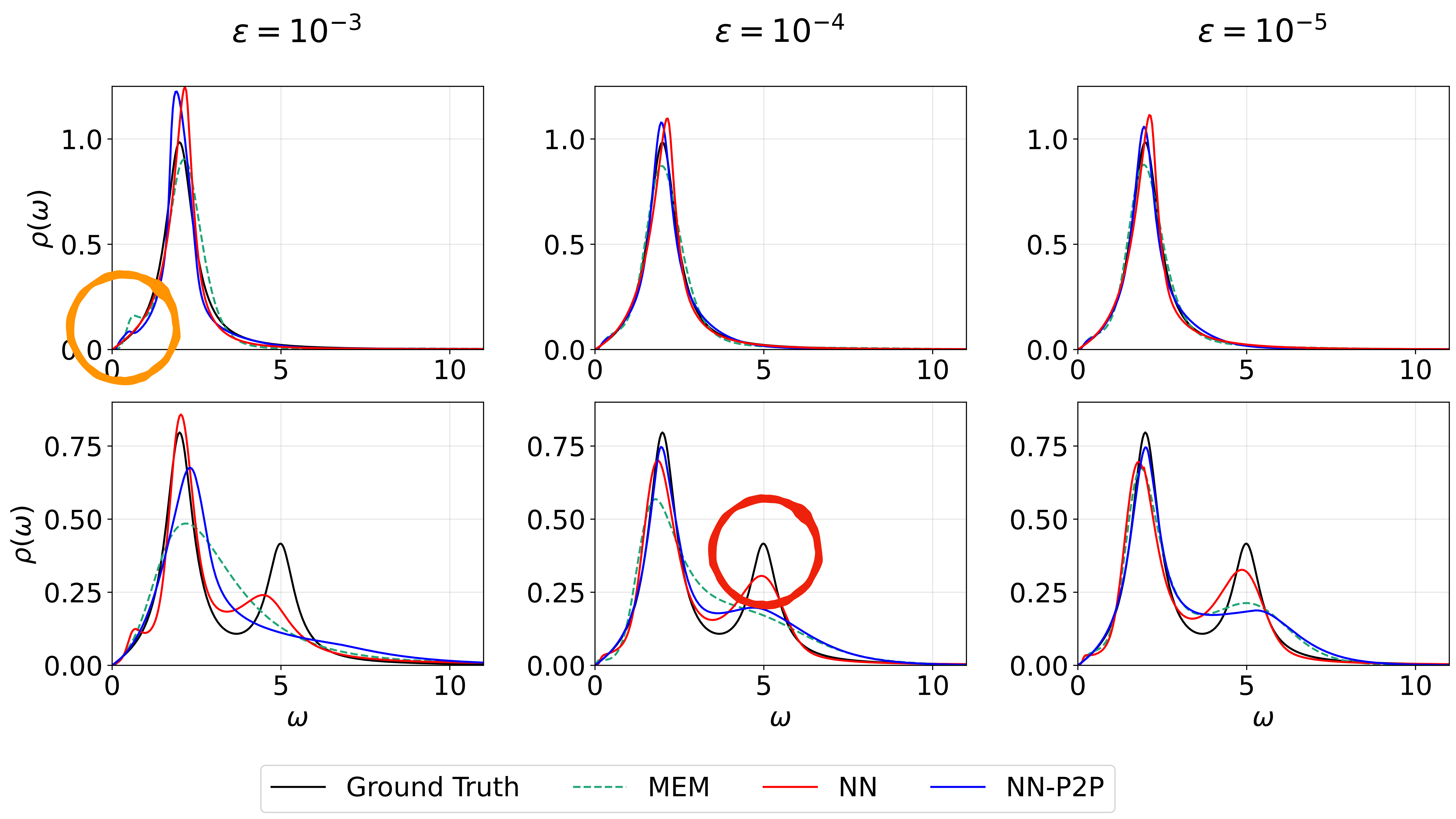
Phys. Rev. D 106, L051502

Mock Test I.

$$\rho^{(\mathrm{BW})}(\omega) = \frac{4A\Gamma\omega}{\left(M^2+\Gamma^2-\omega^2\right)^2+4\Gamma^2\omega^2}$$

$A = 1.0, \Gamma = 0.5, M = 2.0$

$A_1 = 0.8, A_2 = 1.0, \Gamma_1 = \Gamma_2 = 0.5$
$M_1 = 2.0, M_2 = 5.0$



noise level ϵ
$N_p = 25$ points

Physics-Driven Deep Learning

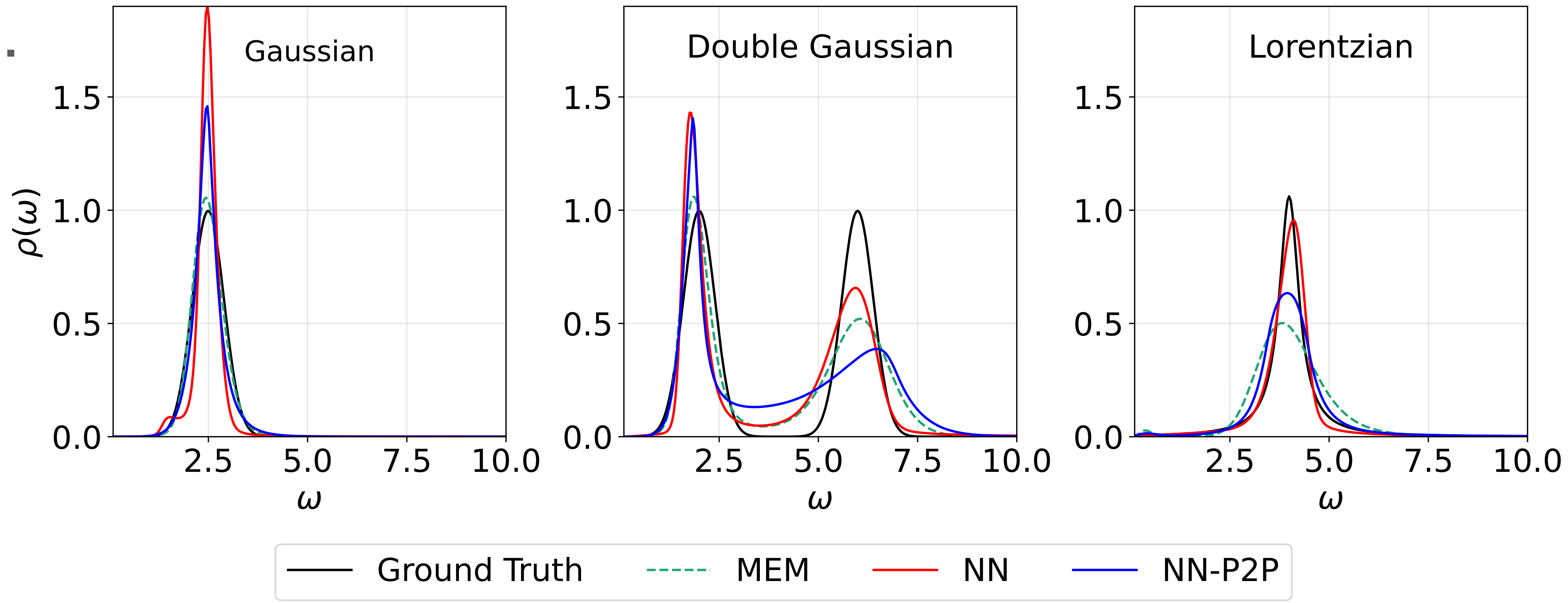
2. Reconstructing Spectral Function

Phys. Rev. D 106, L051502

Mock Test II.

noise level $\epsilon = 10^{-4}$ with $N_p = 25$ points

1.



1. Single-peak functions

2. Non-positive-definited SPs

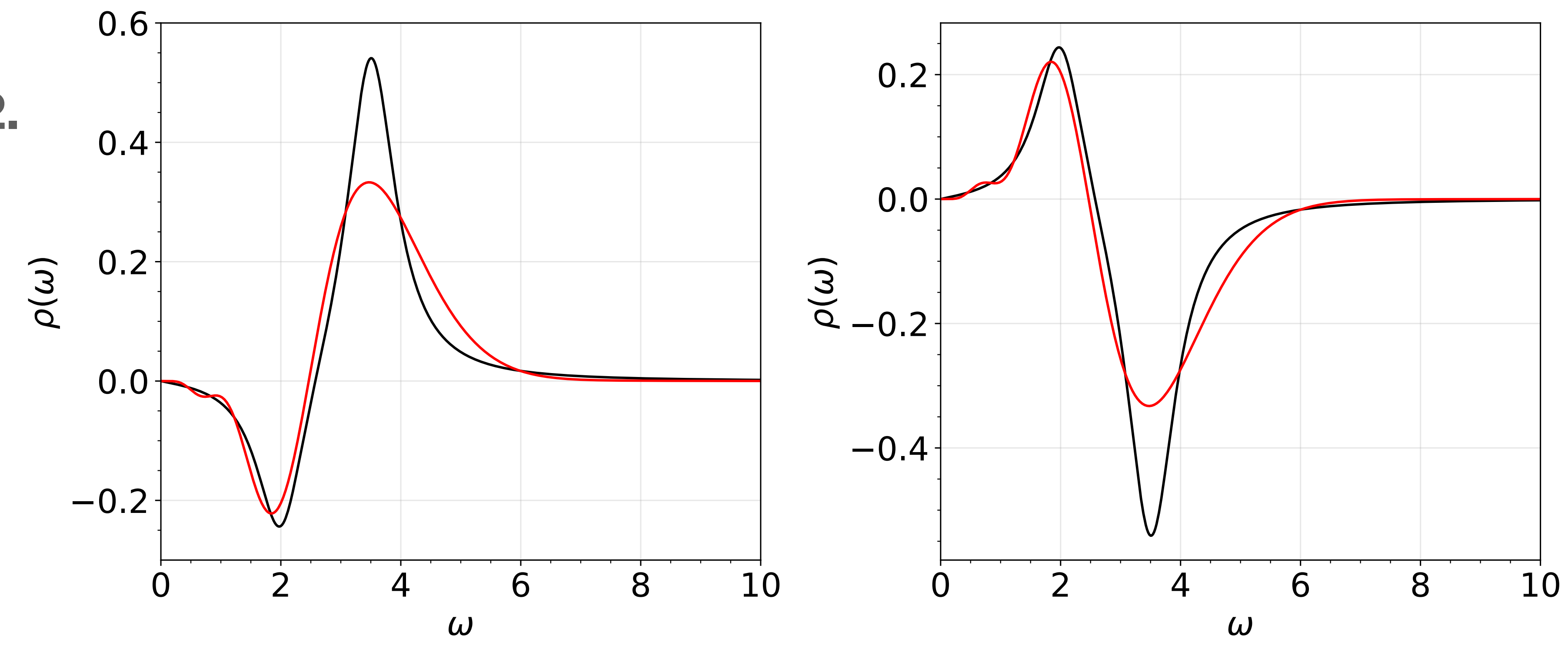
3. Lattice QCD mock data

Thermal (details see arXiv:2110.13521)

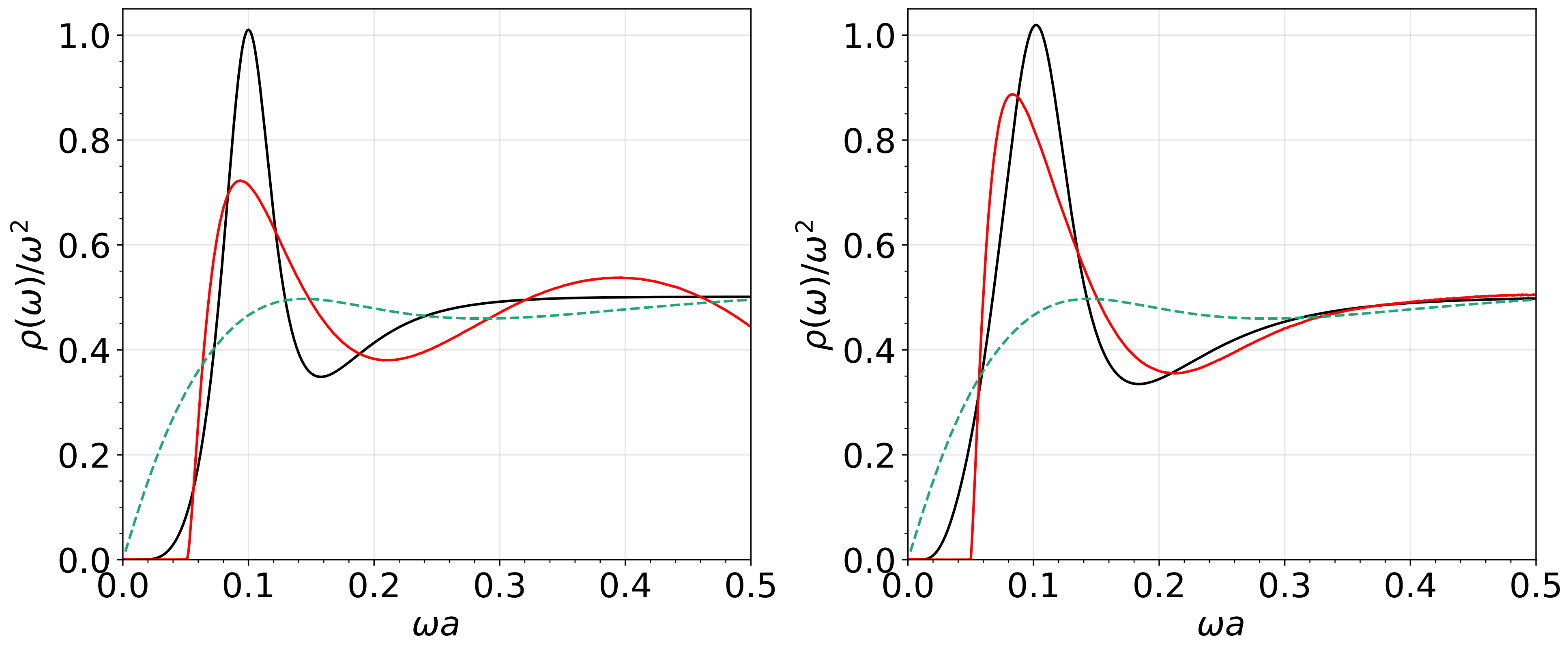
$$G(\tau, T) = \int_0^\infty \frac{d\omega}{2\pi} K(\omega, \tau, T)\rho(\omega, T)$$

$$K(\omega, \tau, T) = \frac{\cosh \omega(\tau - \frac{1}{2T})}{\sinh \frac{\omega}{2T}}$$

2.



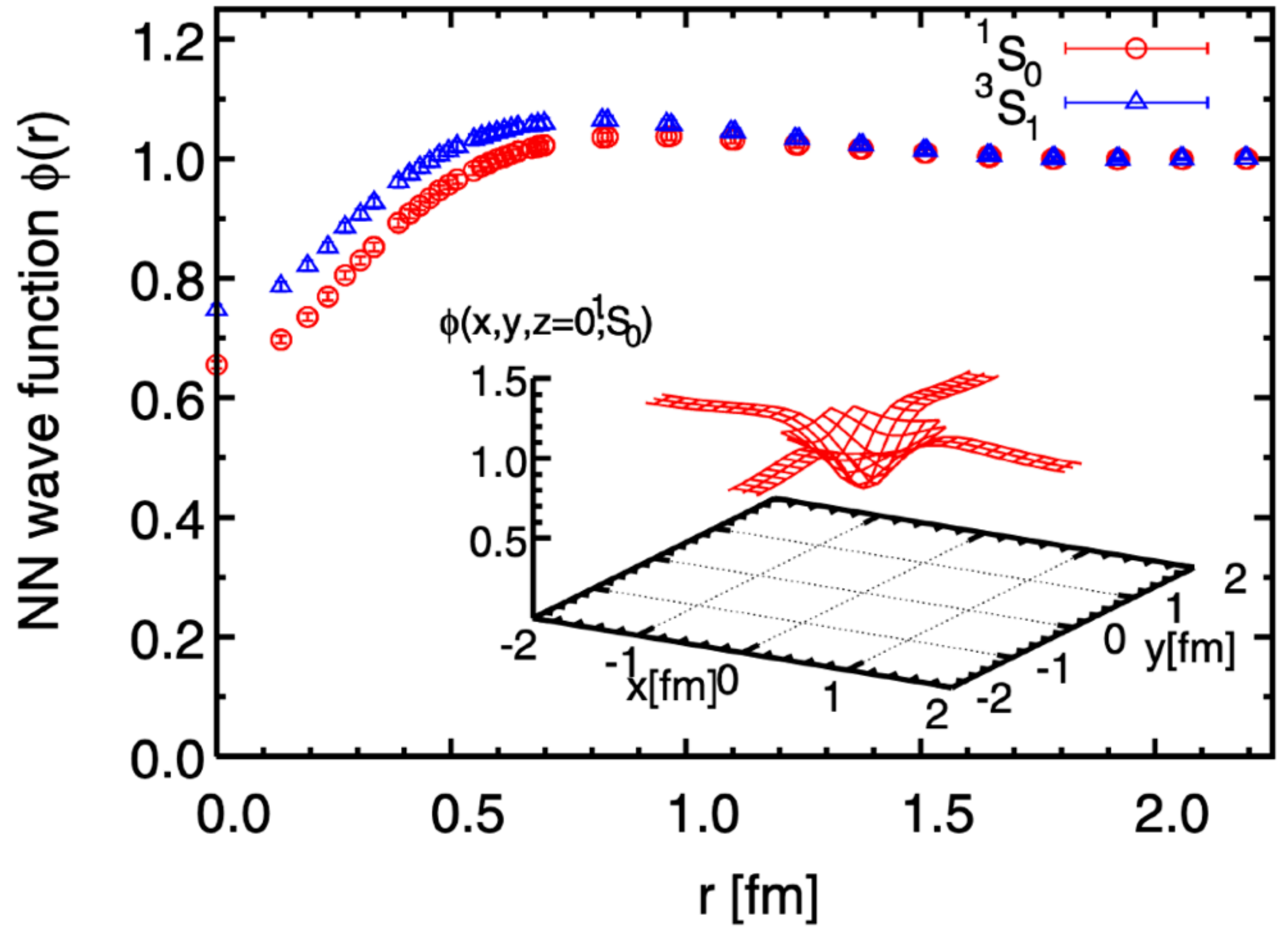
3.

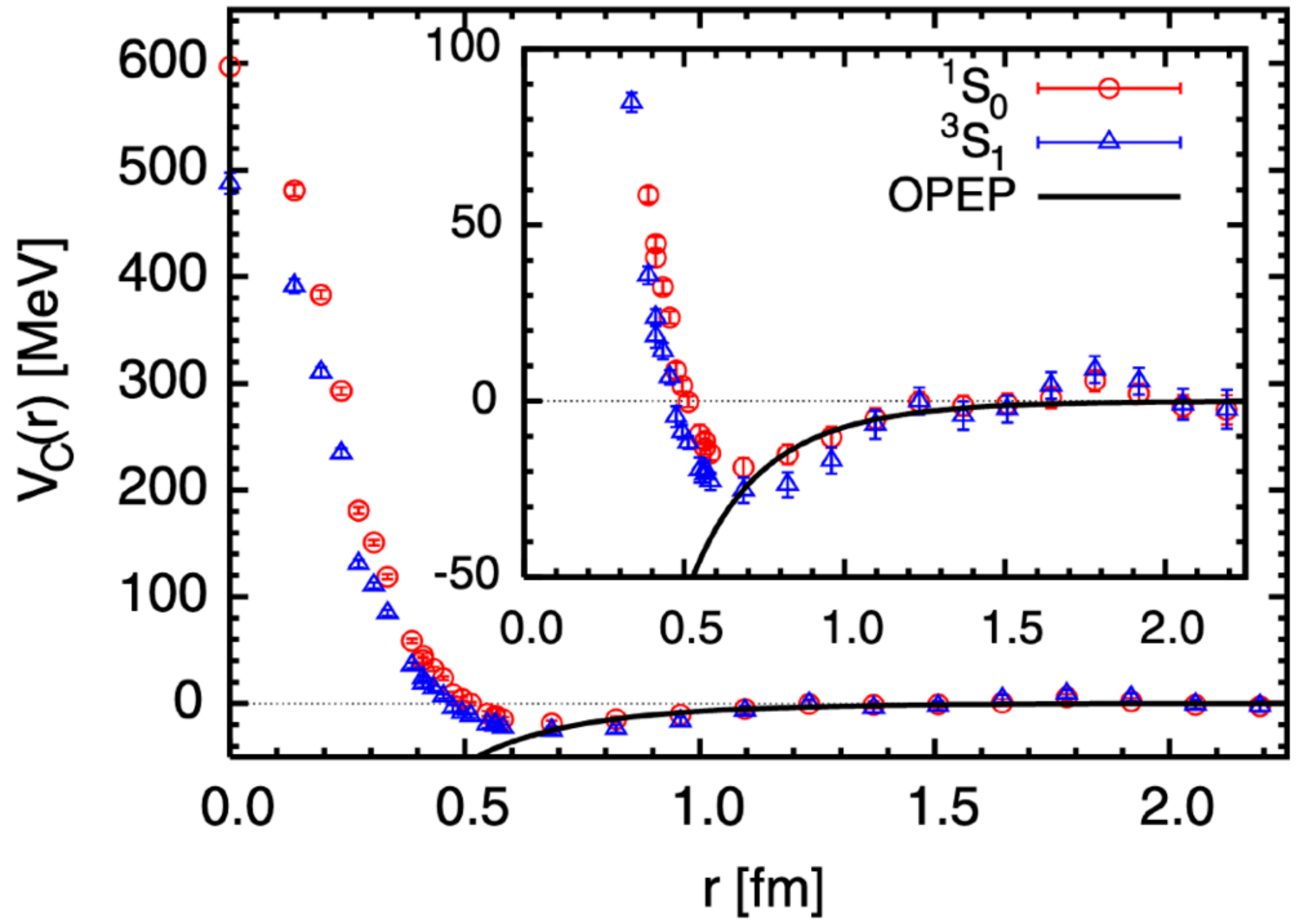


Physics-Driven Deep Learning

3. Extracting Nuclear Force

N. Ishii, S. Aoki, and T. Hatsuda, Phys. Rev. Lett. **99**, 022001 (2007)





Nambu-Bethe-Salpeter (NBS) wave function

$$\psi_{NBS}(\vec{r}) = \langle 0|N(\vec{r})N(\vec{0})|N(\vec{k})N(-\vec{k}), in\rangle$$
$$\simeq e^{i\delta_l(k)}\sin(kr - l\pi/2 + \delta_l(k))/(kr)$$

(at asymptotic region)

Local Approx.
Gradient Expansion

HAL QCD method

Nulcear Force

$$(k^2/m_N - H_0)\,\psi_{NBS}(\vec{r}) = \int d\vec{r}'\,U(\vec{r},\vec{r}')\psi_{NBS}(\vec{r}')$$

(Schrodinger eq.)

Physics-Driven Deep Learning

3. Extracting Nuclear Force

in preparation (with HAL QCD)

Two identical-particle interactions

$U_\theta\,(r, r')$

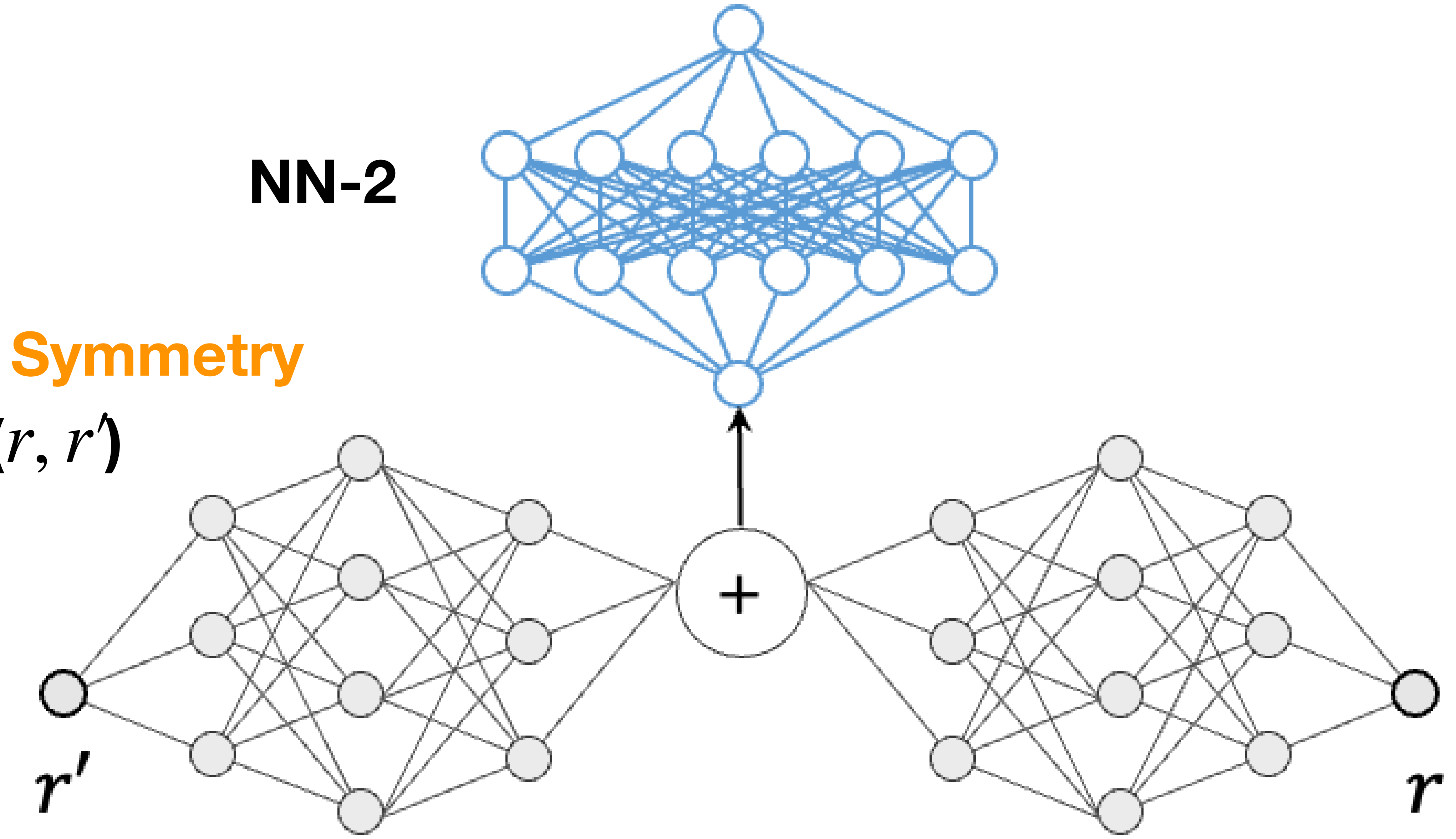
BP

Residual of Schrödinger Eq.

NN-2

Exchange Symmetry

for (r, r')



NN-1 **NN-1**

$$(E_k - H_0)\phi_{\mathbf{k}}(\mathbf{r}) = \int d^3 r' U(\mathbf{r}, \mathbf{r}')\phi_{\mathbf{k}}(\mathbf{r}')$$

$$E_k = \frac{k^2}{2m}, \quad H_0 = -\frac{\nabla^2}{2m}, \quad m = \frac{m_N}{2}$$

$\phi_k\,(r)$

or

$R_t\,(r)$

Phys. Lett. B **712**, 437 (2012)

$$\left\{ \frac{1}{4m_N}\frac{\partial^2}{\partial t^2} - \frac{\partial}{\partial t} - H_0 \right\} R(t, r) = \int 4\pi r'^2 dr'\, U(r, r') R(t, r')$$

Physics-Driven Deep Learning

3. Extracting Nuclear Force

in preparation (with HAL QCD)

$\Omega_{ccc}\Omega_{ccc}$ **Interaction**

$$\left\{\frac{1}{4m_N}\frac{\partial^2}{\partial t^2}-\frac{\partial}{\partial t}-H_0\right\}R(t,r)=\int 4\pi r'^2 dr'\,U(r,r')R(t,r')$$

Nambu-Bethe-Salpeter (NBS) wave function

$$R2=R_{t+1}-2R_t+R_{t-1}, R1=(R_{t+1}-R_{t-1})/2, Rr=\nabla^2 R(t,r)$$

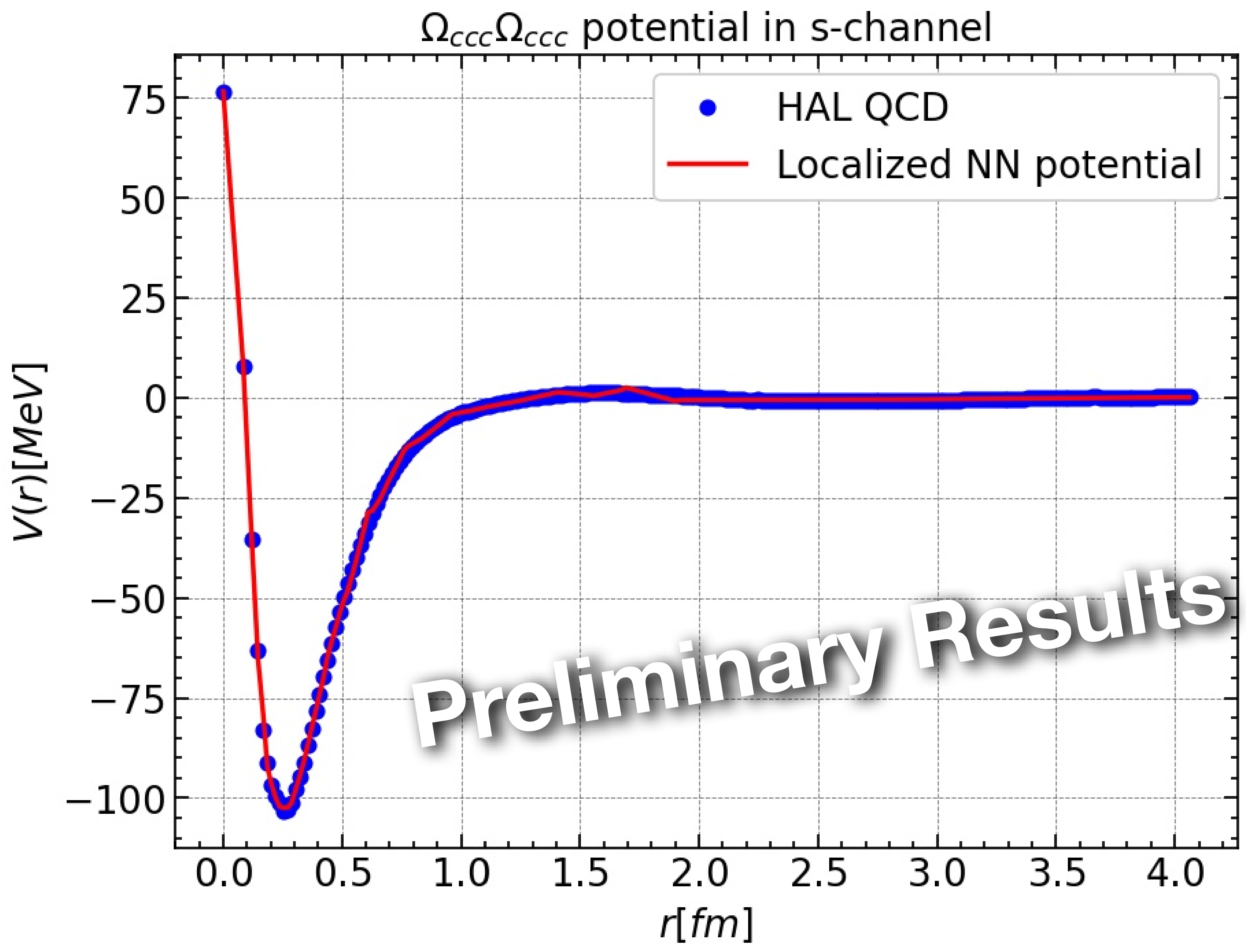
$$m_N=2.073, a^{-1}=2333.0\textbf{MeV}$$

$$t=26$$

Phys. Rev. Lett. 127, 072003 (2021)

No Approx.

Neural Network Hadron Force



$$\mathcal{L}=\sum_t\left\{\frac{1}{4m_N}R2(t,r)-R1(t,r)+\frac{1}{m_N}Rr(t,r)-\int 4\pi r'^2 dr'\,U_\theta(r,r')R(t,r')\right\}$$

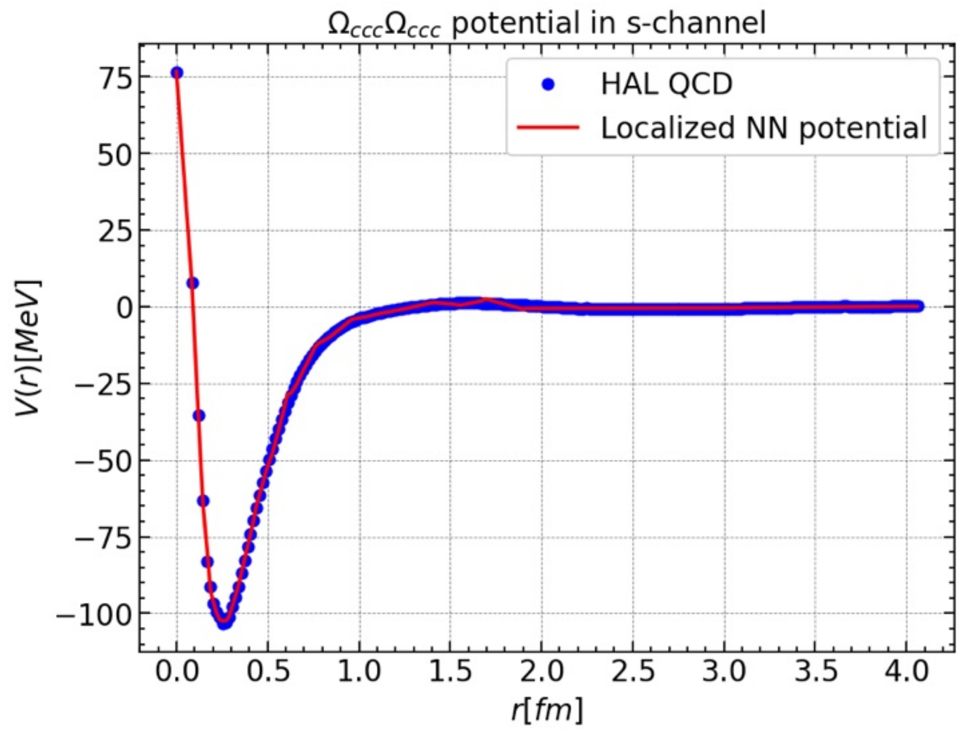
$$V_\theta(r)\equiv\frac{\sum_{r'}\Delta r'\,U_\theta(r,r')R(t,r')}{R(t,r)}$$

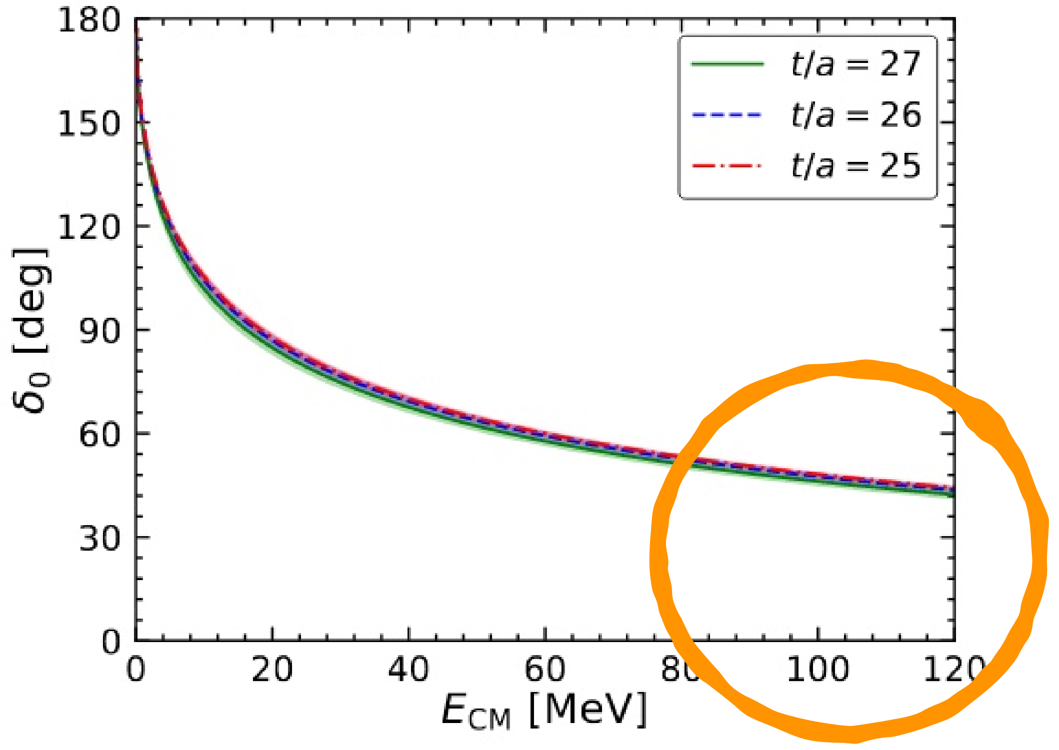
Physics-Driven Deep Learning

3. Extracting Nuclear Force

in preparation (with HAL QCD)

$\Omega_{ccc}\Omega_{ccc}$ **Interaction**



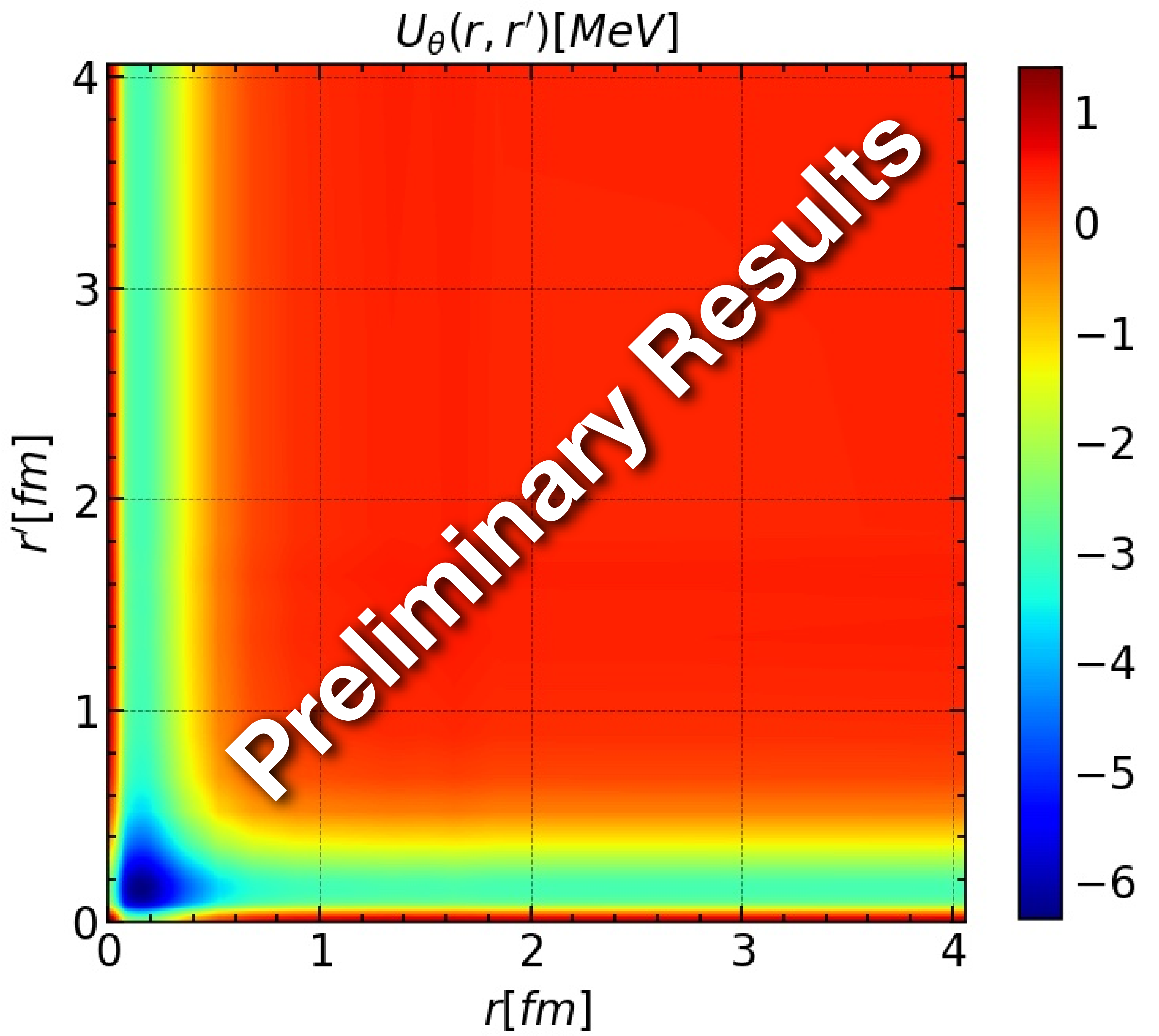


$$(E_k - H_0)\phi_{\mathbf{k}}(\mathbf{r}) = \int d^3 r' U_\theta(\mathbf{r}, \mathbf{r}')\phi_{\mathbf{k}}(\mathbf{r}')$$

$$E_k = \frac{k^2}{2m}, \quad H_0 = -\frac{\nabla^2}{2m}, \quad m = \frac{m_N}{2}$$

← **Affect**

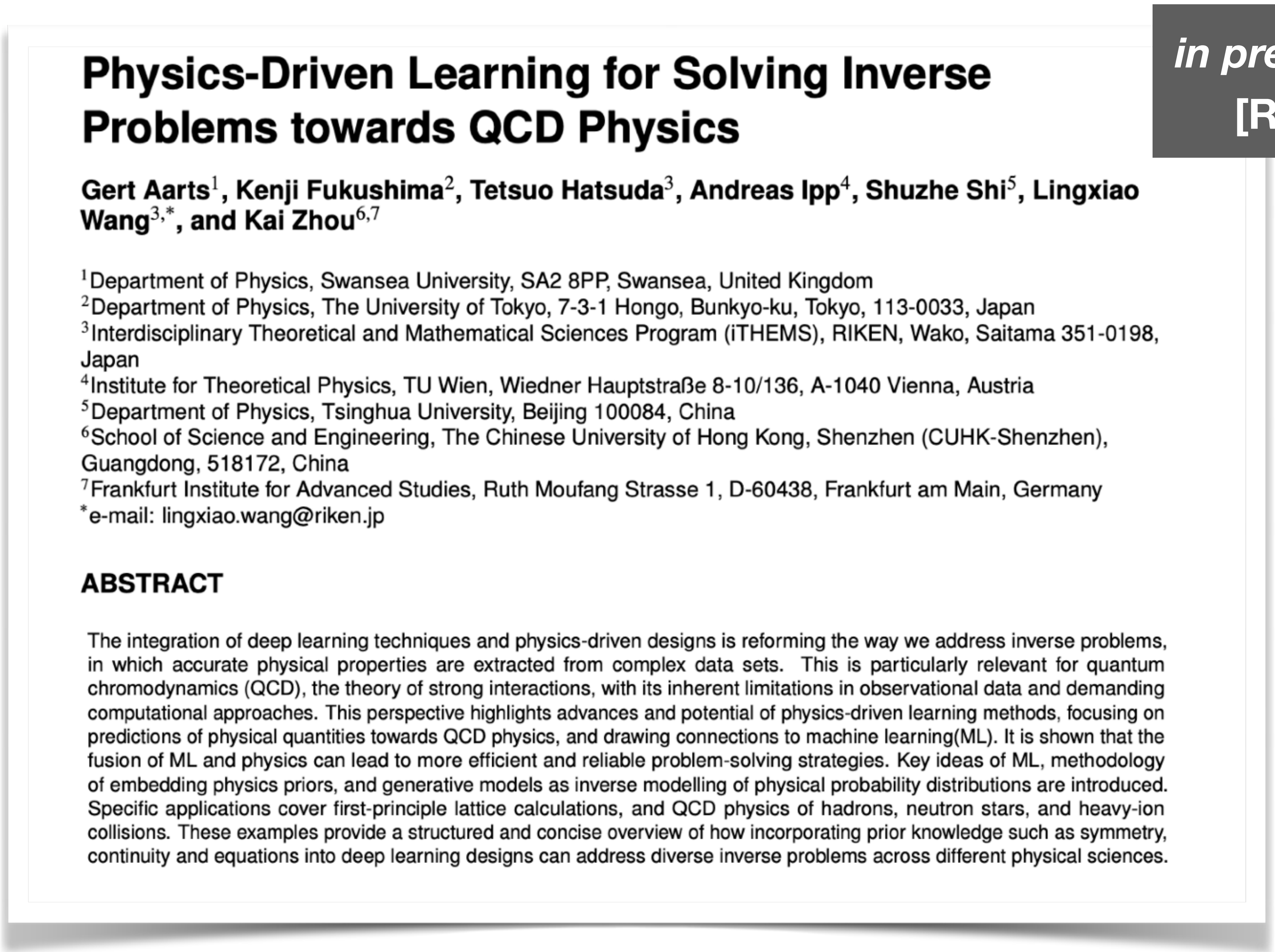
Neural Network Hadron Force



Summary I

- **Inverse Problems**
  - Data-driven learning
  - Physics-driven learning
    - Neural network representations
    - Embed physics priors explicitly
      - **Continuity** and **Causality** for EoSs
      - **Continuity** and **Positive Defenetion** for SFs
      - **Exchange Symmetry** and **Asymptotic Behavior** for HH interactions
- **Future works**
  - PTs in Nuclear Matter EoS
  - More Spectroscopy [github1, github2]
  - More Hadron Forces

in preparation **[Review]**



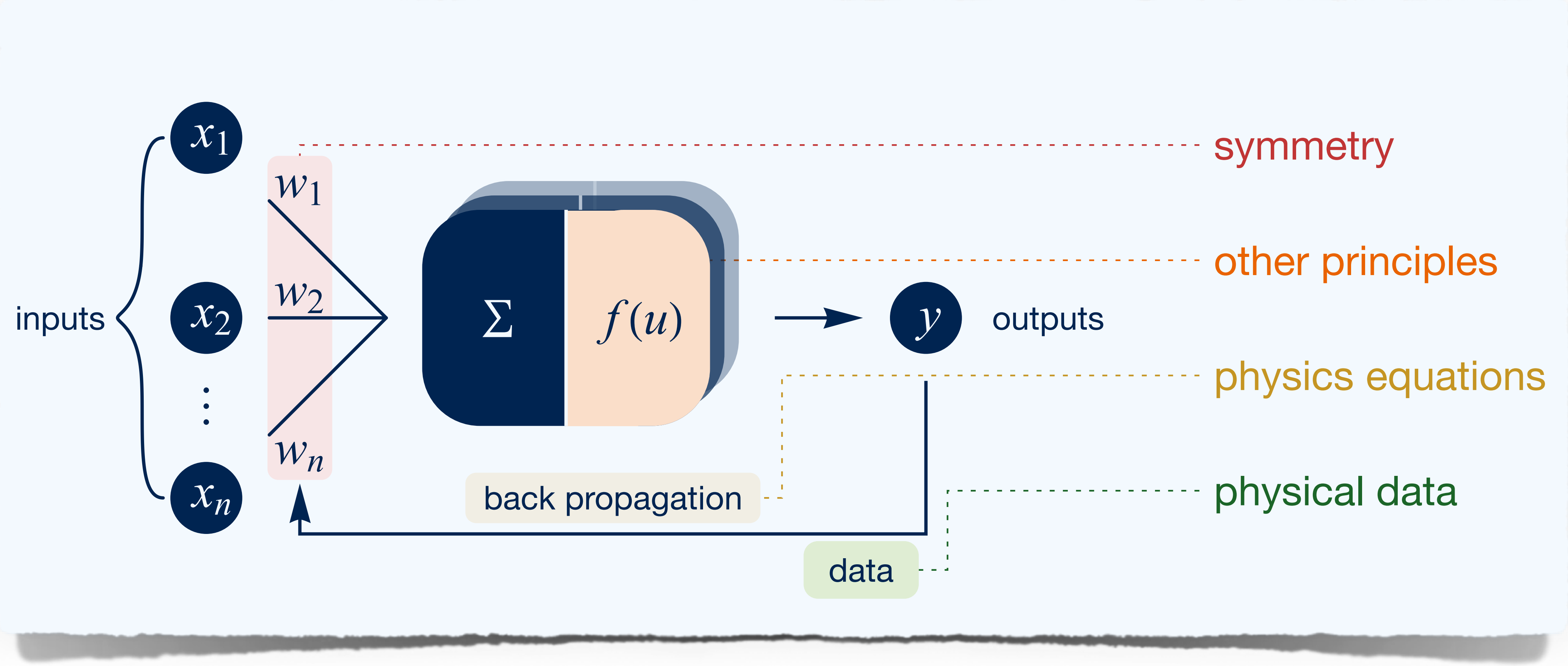
Physics-Driven Learning for Solving Inverse Problems towards QCD Physics

Gert Aarts[1], Kenji Fukushima[2], Tetsuo Hatsuda[3], Andreas Ipp[4], Shuzhe Shi[5], Lingxiao Wang[3,*], and Kai Zhou[6,7]

[1]Department of Physics, Swansea University, SA2 8PP, Swansea, United Kingdom
[2]Department of Physics, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-0033, Japan
[3]Interdisciplinary Theoretical and Mathematical Sciences Program (iTHEMS), RIKEN, Wako, Saitama 351-0198, Japan
[4]Institute for Theoretical Physics, TU Wien, Wiedner Hauptstraße 8-10/136, A-1040 Vienna, Austria
[5]Department of Physics, Tsinghua University, Beijing 100084, China
[6]School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen (CUHK-Shenzhen), Guangdong, 518172, China
[7]Frankfurt Institute for Advanced Studies, Ruth Moufang Strasse 1, D-60438, Frankfurt am Main, Germany
*e-mail: lingxiao.wang@riken.jp

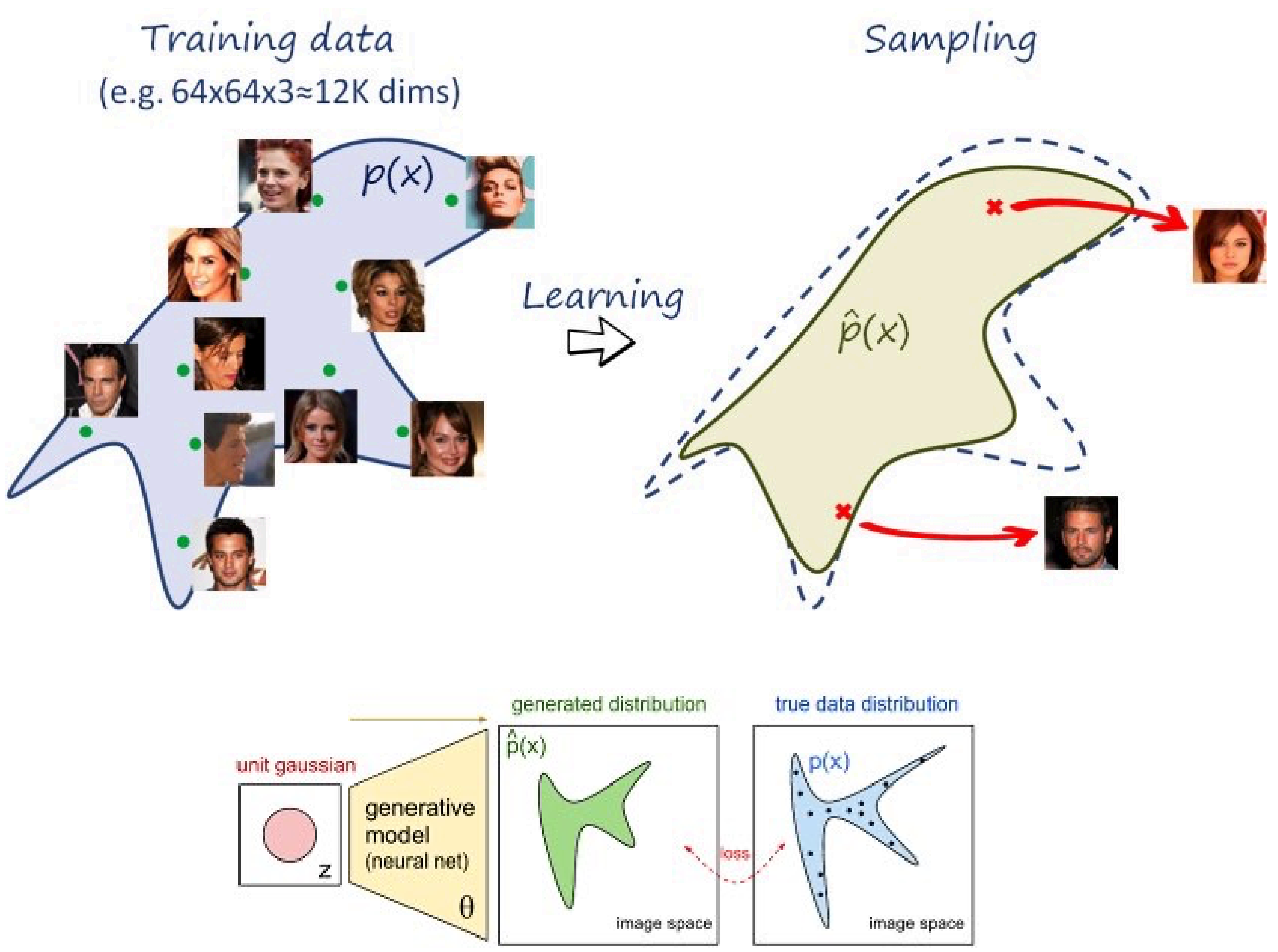
ABSTRACT

The integration of deep learning techniques and physics-driven designs is reforming the way we address inverse problems, in which accurate physical properties are extracted from complex data sets. This is particularly relevant for quantum chromodynamics (QCD), the theory of strong interactions, with its inherent limitations in observational data and demanding computational approaches. This perspective highlights advances and potential of physics-driven learning methods, focusing on predictions of physical quantities towards QCD physics, and drawing connections to machine learning(ML). It is shown that the fusion of ML and physics can lead to more efficient and reliable problem-solving strategies. Key ideas of ML, methodology of embedding physics priors, and generative models as inverse modelling of physical probability distributions are introduced. Specific applications cover first-principle lattice calculations, and QCD physics of hadrons, neutron stars, and heavy-ion collisions. These examples provide a structured and concise overview of how incorporating prior knowledge such as symmetry, continuity and equations into deep learning designs can address diverse inverse problems across different physical sciences.

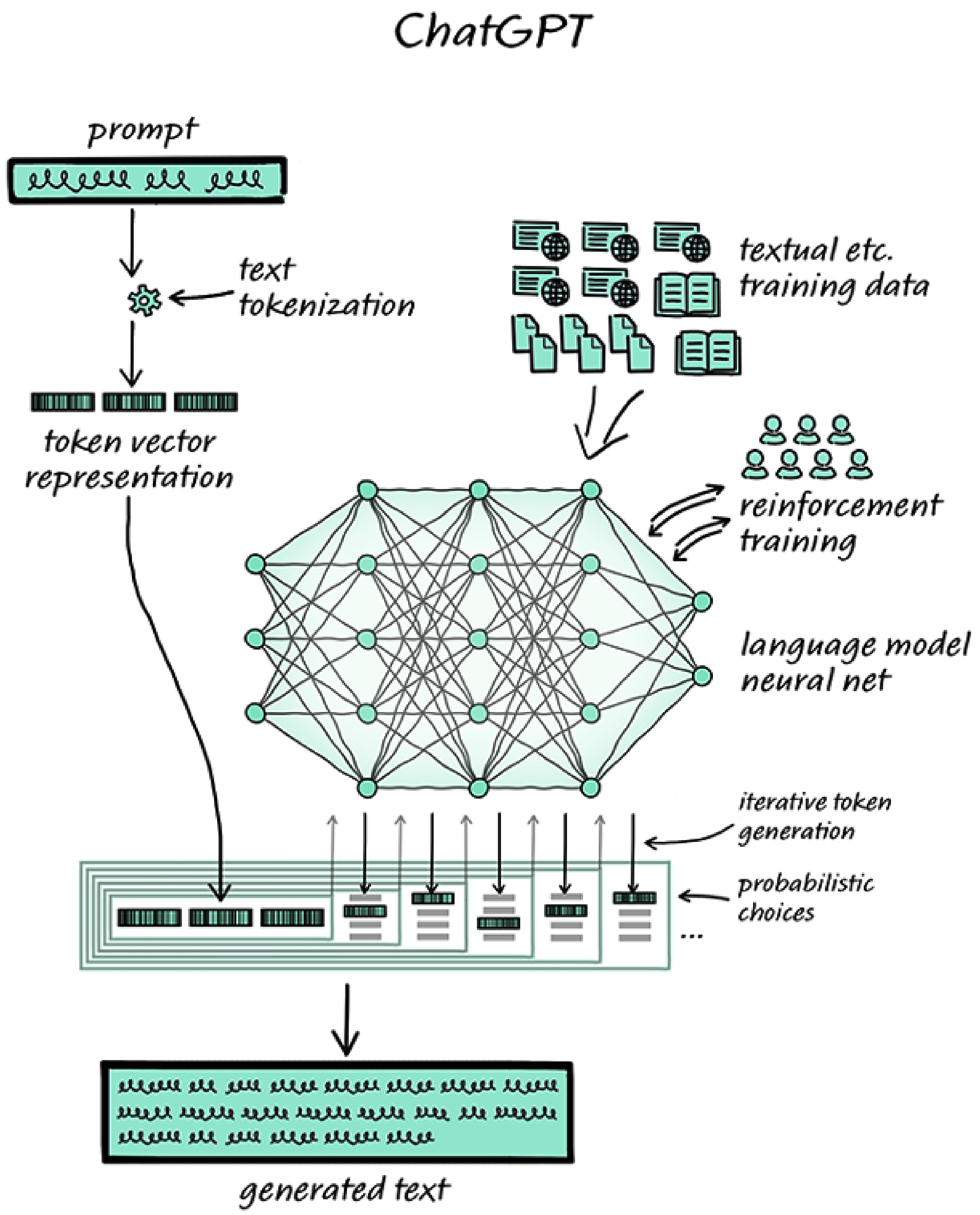


Generative Models
as Inverse Modeling

Generative Models



@blogs of OpenAI



Generative models

→ **Underlying Distributions in Data**

$$\max_{\theta} \prod_{i=1}^{N} p_\theta(\mathbf{x}_i)$$

High-Dimensional Distribution

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

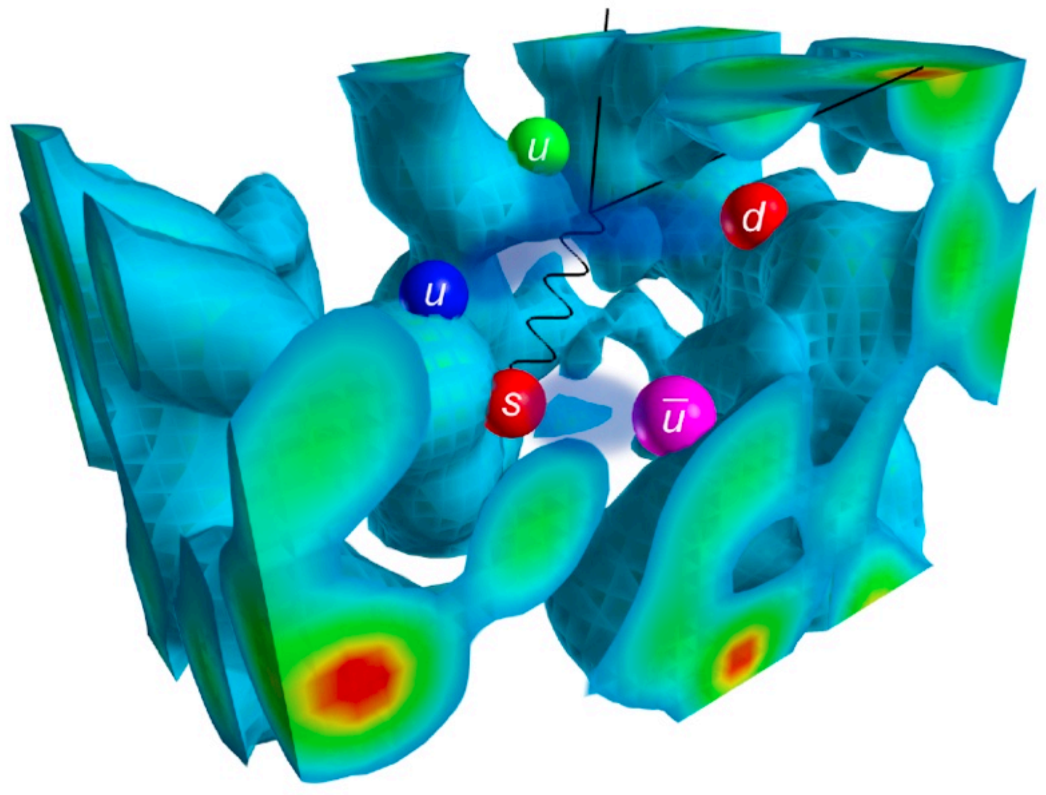
$$\langle O \rangle \approx \frac{1}{N} \sum_i O_i$$

→ **Physical Distribution, Sampling** via Generative Models

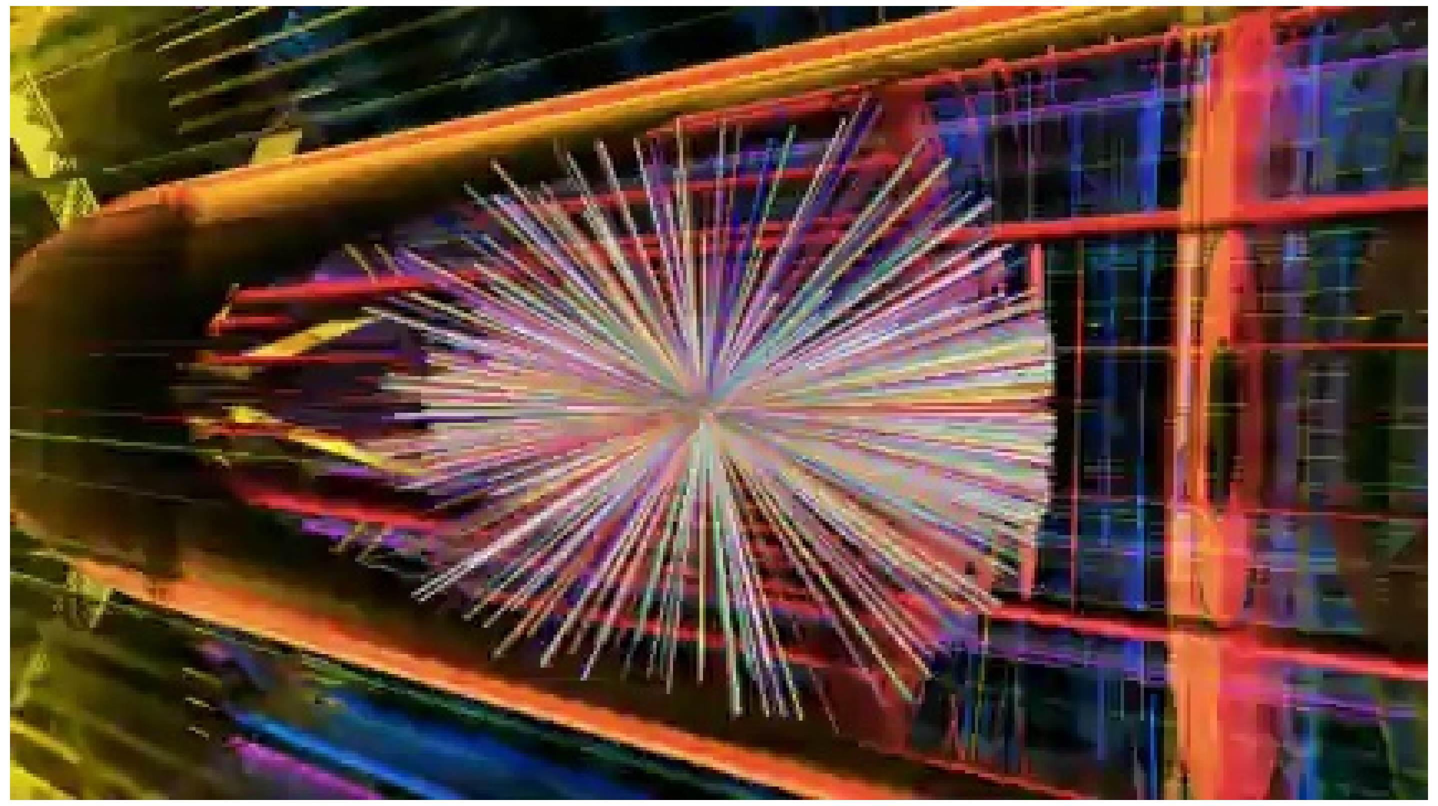
Global Sampling

Fast and Independent Sampler

Prog.Part.Nucl.Phys. 104084(2023)



Lattice QCD © Derek Leinweber/CSSM/University of Adelaide



Heavy-Ion Collisions © 2010 CERN

Learn to Sample

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

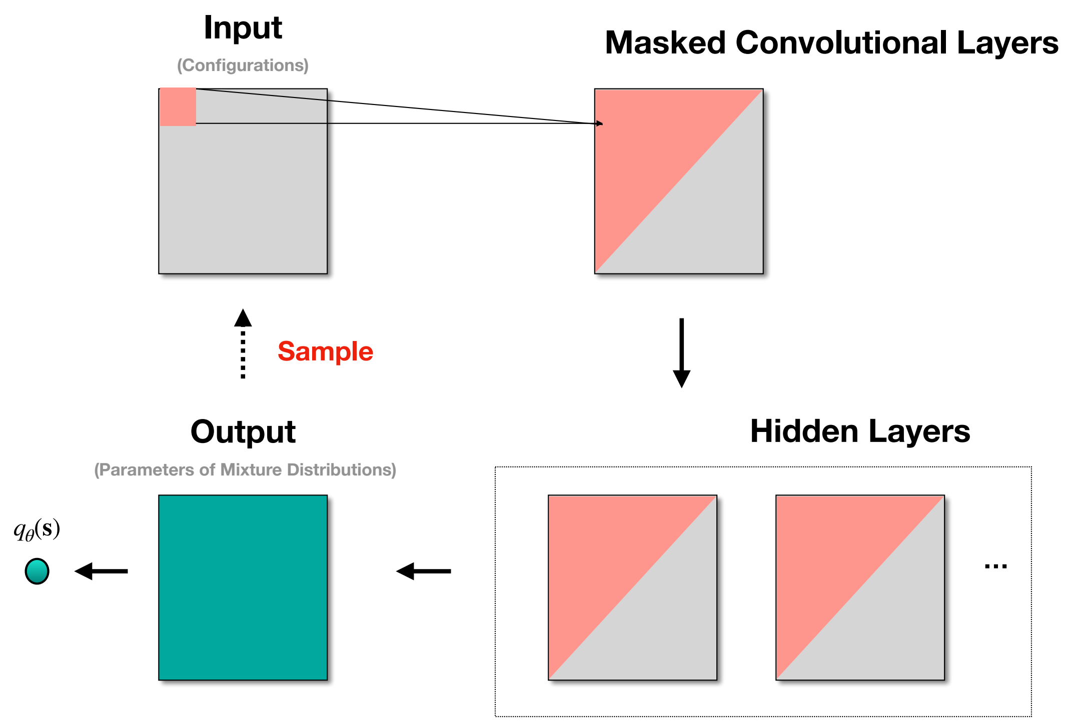
$$\langle O \rangle \approx \frac{1}{N}\sum_i O_i$$

→ **Physical Distribution, Sampling**
via Generative Models

Global Sampling

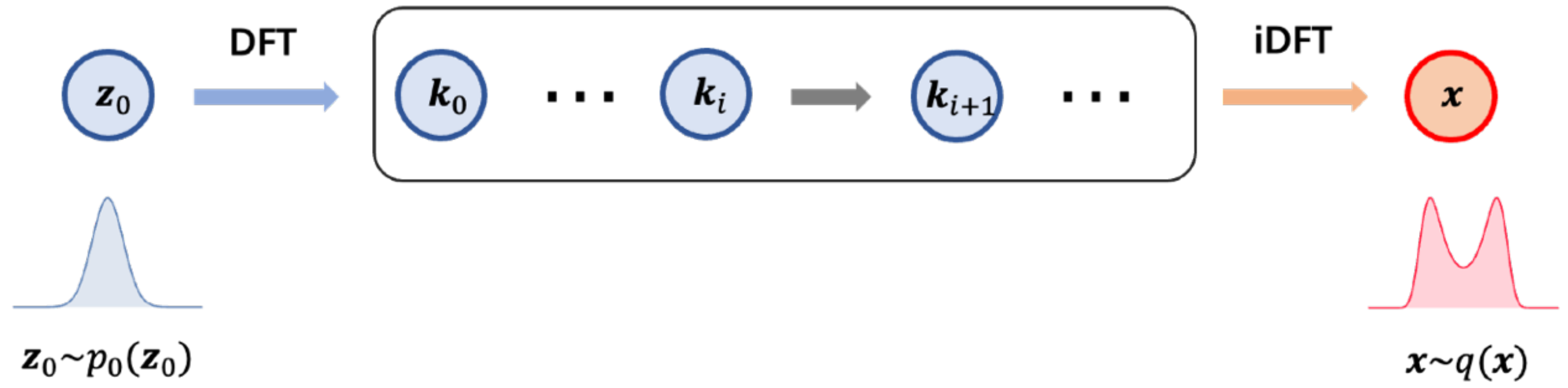
Fast and Independent Sampler

Prog.Part.Nucl.Phys. 104084(2023)



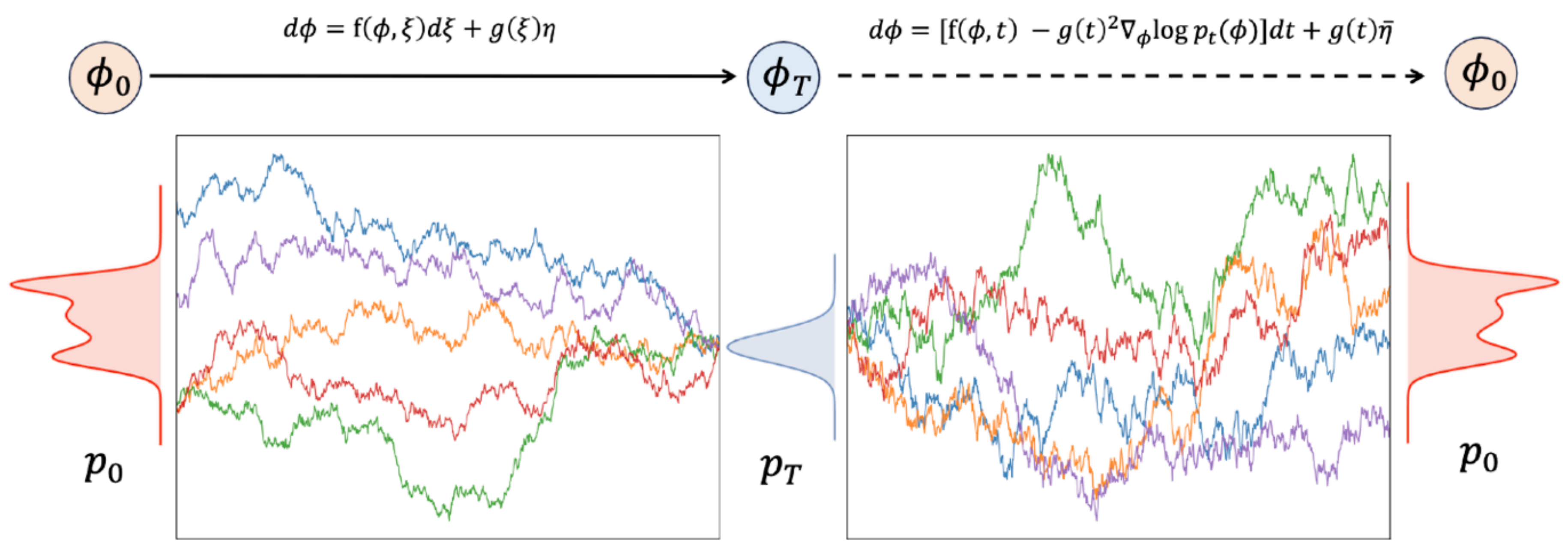
Continuous Autoregressive Models

Chinese Phys. Lett. 39, 120502 (2022)



Fourier Flow-based Model

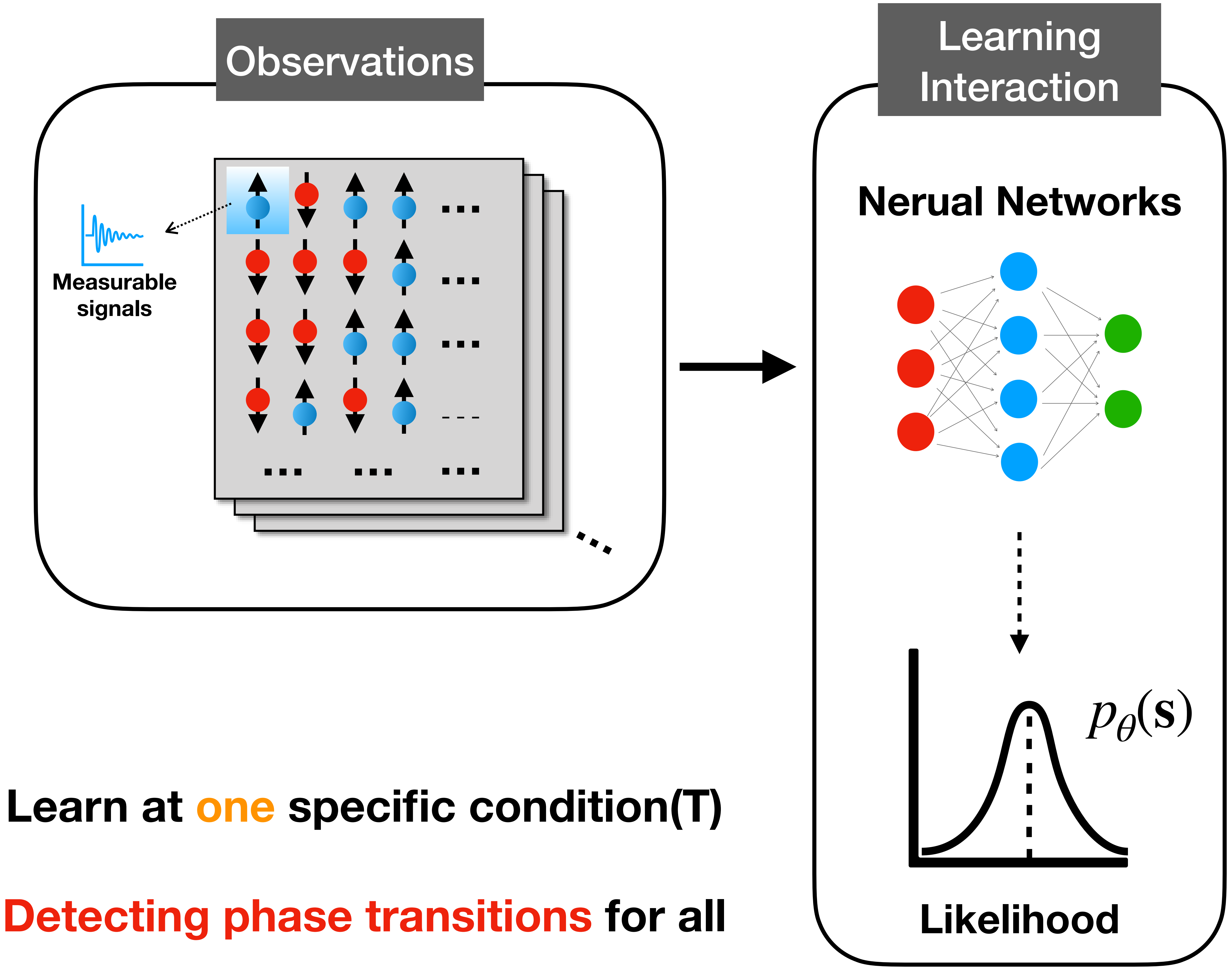
Phys. Rev. D 107, 056001



Diffusion Models

JHEP 05(2024)060

Learn Micro-Interactions from Observations



Learn at one specific condition(T)

Detecting phase transitions for all

$$\max_{\theta} \prod_{i=1}^{N} p_\theta(\mathbf{s}_i)$$

$$p(\mathbf{s}) \leftarrow p_\theta(\mathbf{s}) \equiv \frac{e^{-\frac{H_\theta(\mathbf{s})}{T}}}{Z}$$

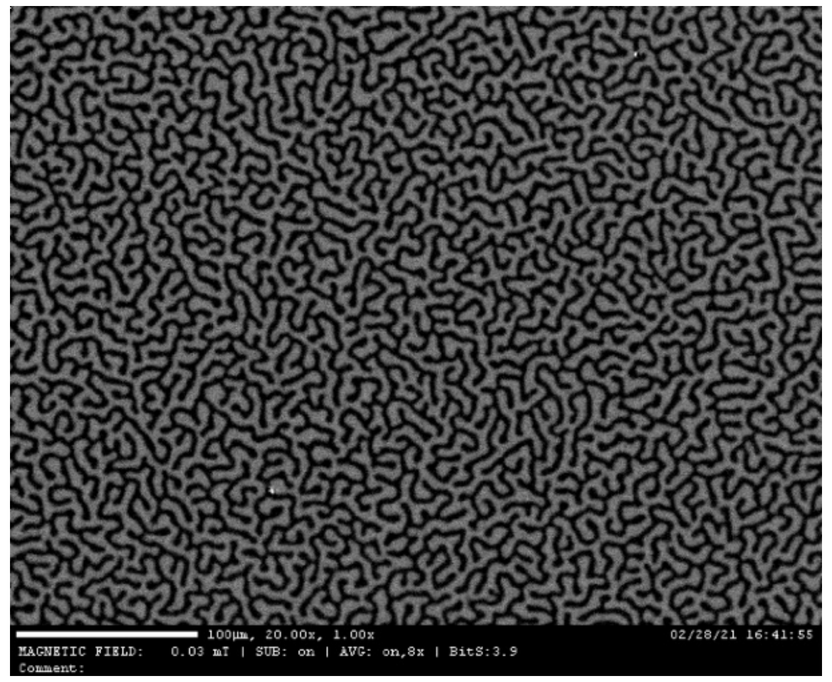
$$H_\theta(\mathbf{s}, T) = -T \ln p_\theta(\mathbf{s}) - T \ln Z$$

$$\frac{\Delta H_\theta(\mathbf{s}, T)}{T'} \equiv -\frac{T}{T'}(\ln p_\theta(\mathbf{s} + \delta\mathbf{s}) - \ln p_\theta(\mathbf{s}))$$

Learn to Detect Phase Transitions

Ferromagnetic Materials

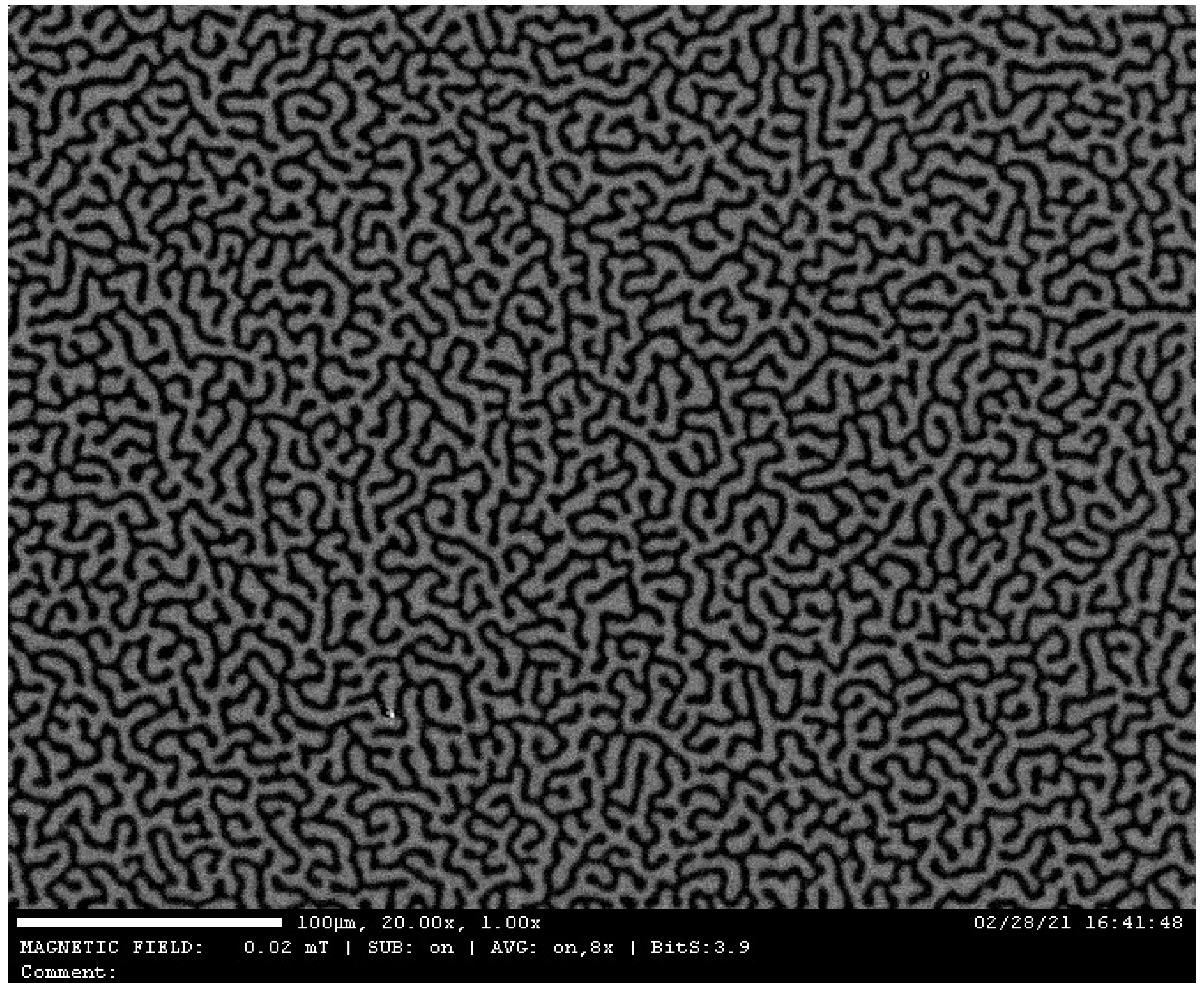
In preparation

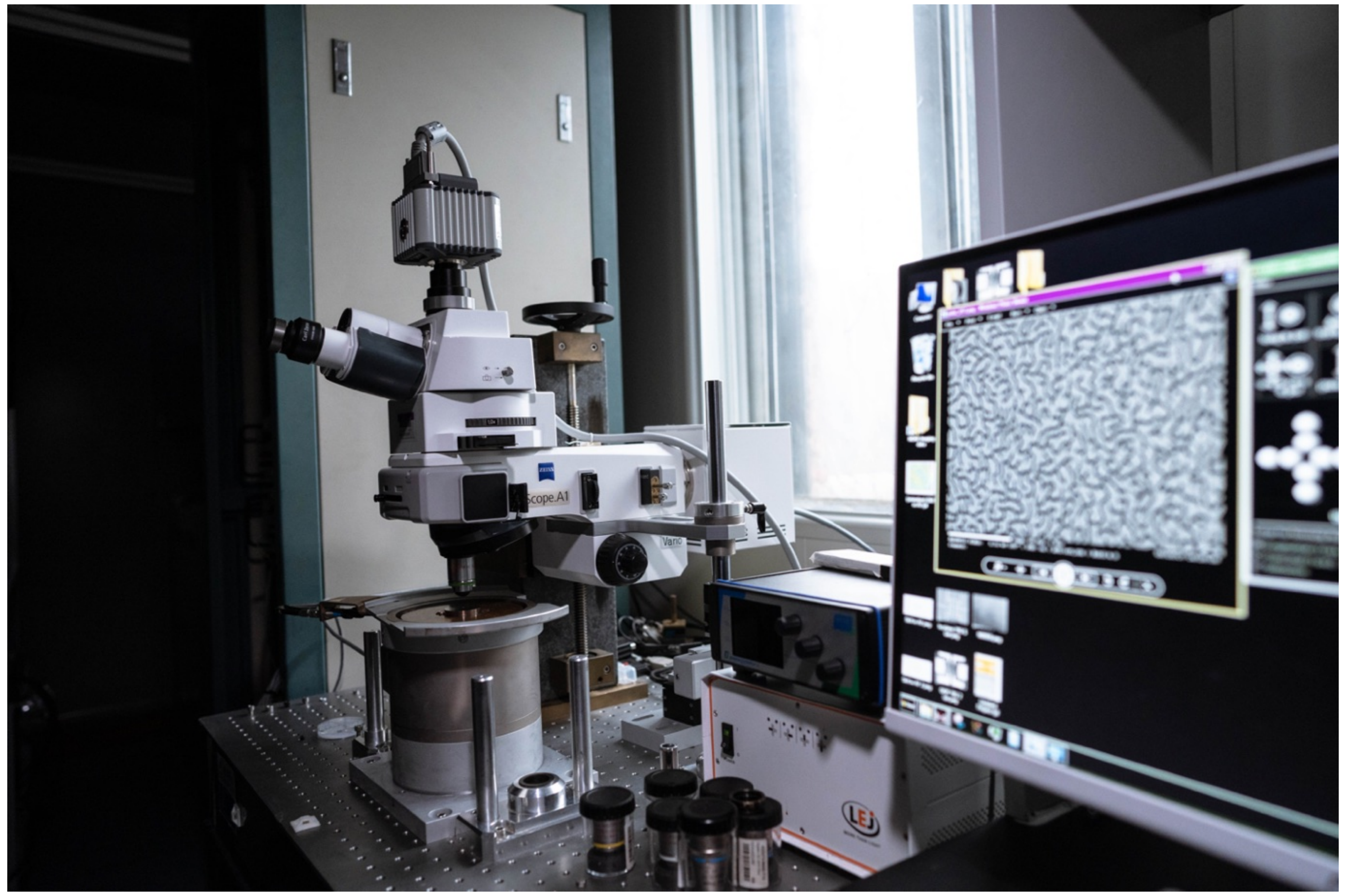


...

3000 images

Resolution: 0.5 μm





PhysRevLett.125.027206

MOKE microscope@THU

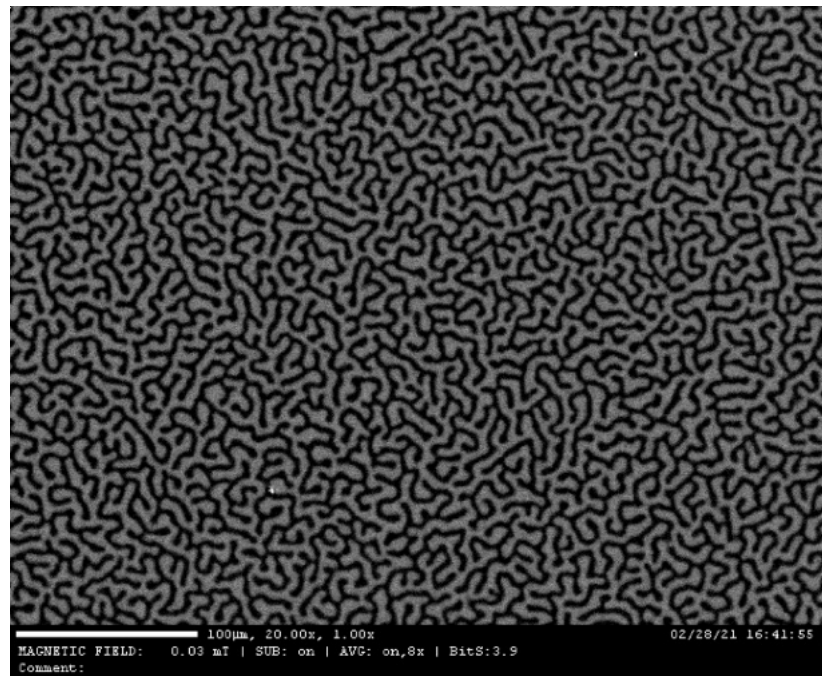
with Le Zhao and Wan-Jun Jiang

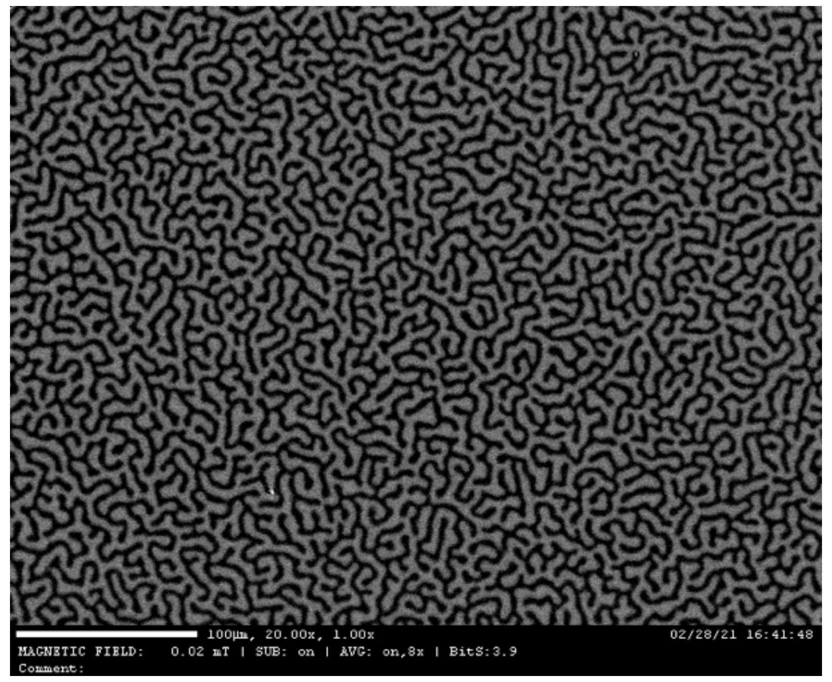
Magneto-optic Kerr effect (MOKE) microscope to capture images
for the **magnetic domains** appearing inside a **Ta/CoFeB/TaO_x thin film**
at **room temperature T = 296 K**

Learn to Detect Phase Transitions

Ferromagnetic Materials

In preparation





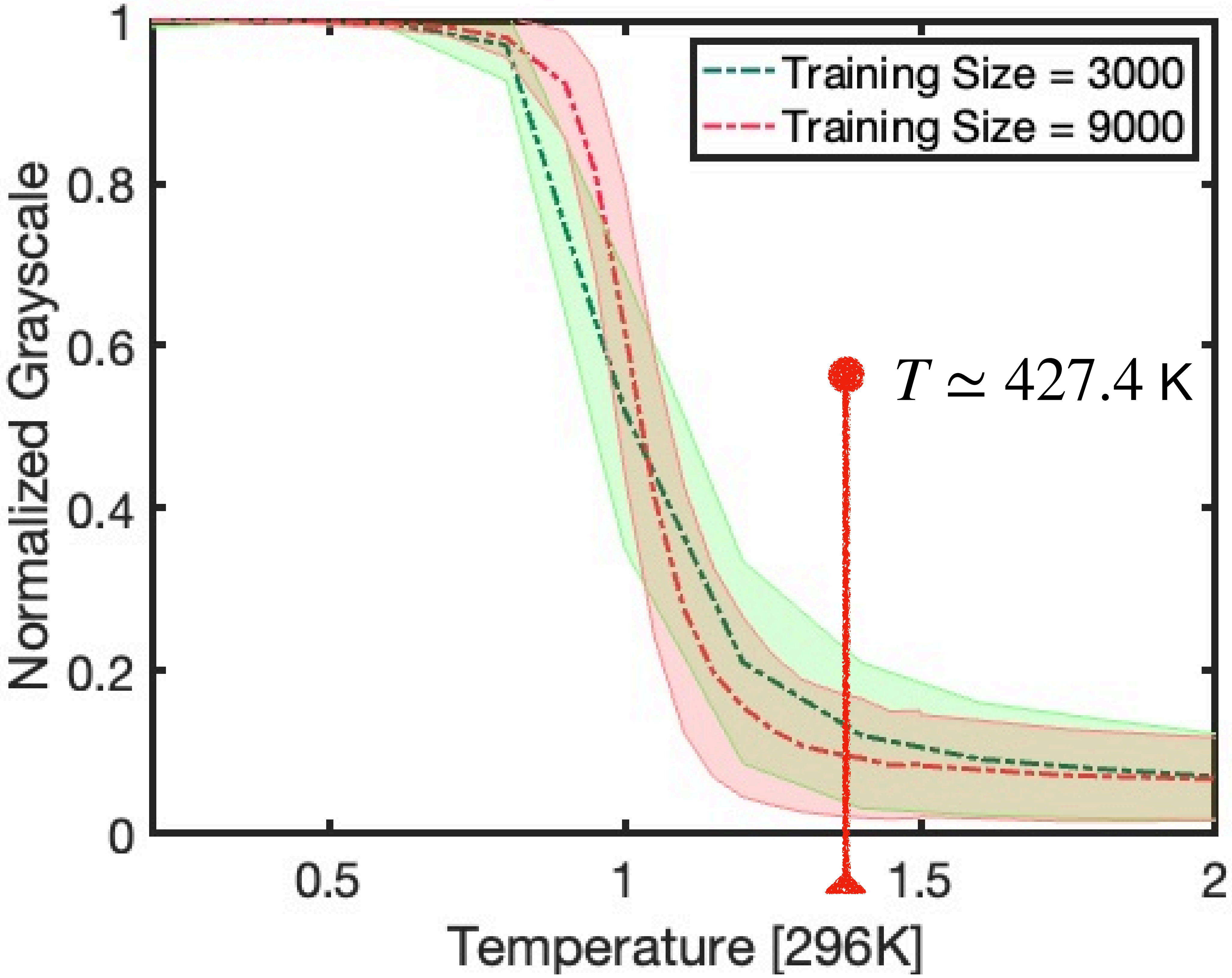
PhysRevLett.125.027206

3000 images

T [K]	50	100	150	200	250	300
M_s [emu/cc]	961.16	925.25	877.73	818.17	753.31	673.427

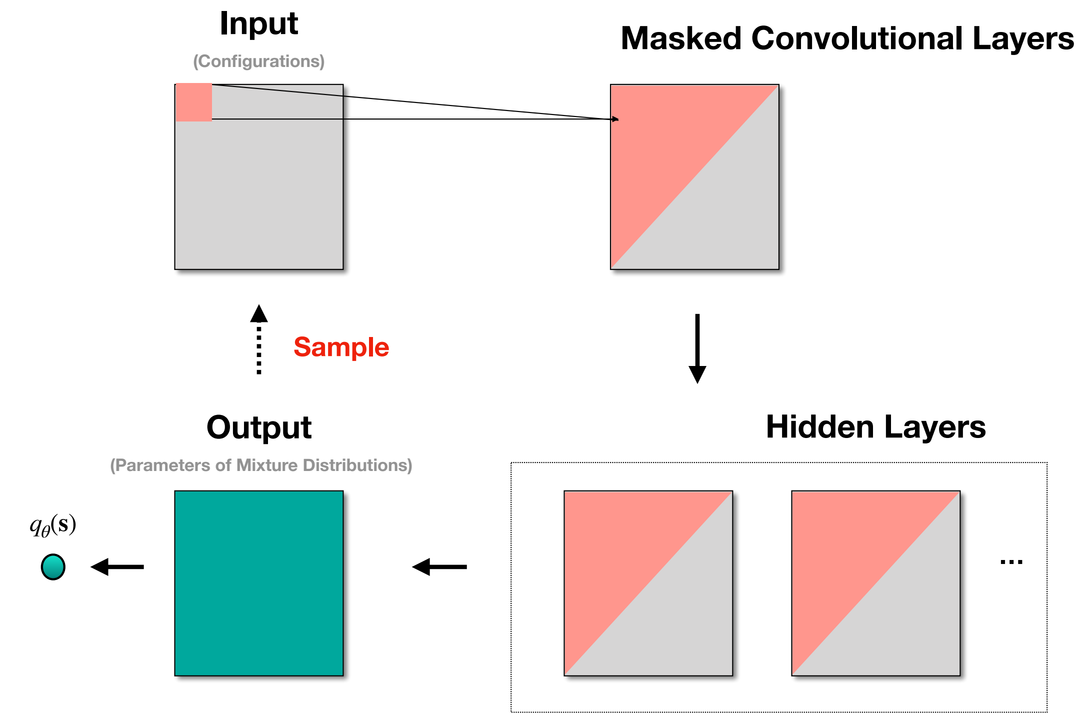
$$M_s(T) = M_s(0)(1 - T/T_C)^{1/3}$$

$$T_C = 427.4 \pm 2.9 \text{ K}$$



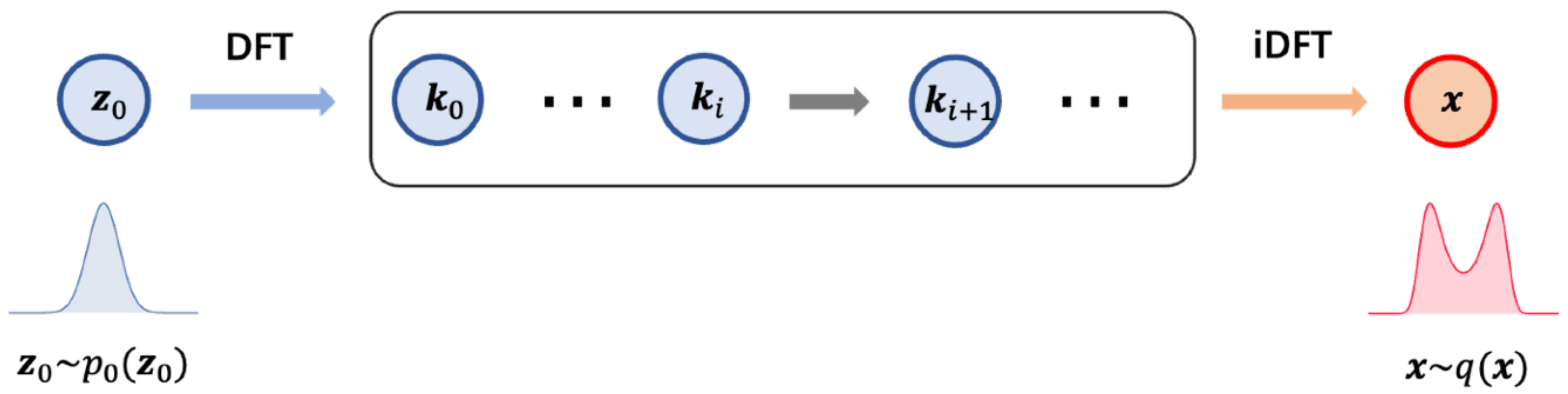
Summary II

- **Generative Models**
  - Learn to Sample
  - Probabilistic Models
  - Learn to Detect Phase Transition
- **Future works**
  - 2D U(1) **Gauge Field**
  - Complex Langevin Method(CLM)
  - Finite Chemical Potential

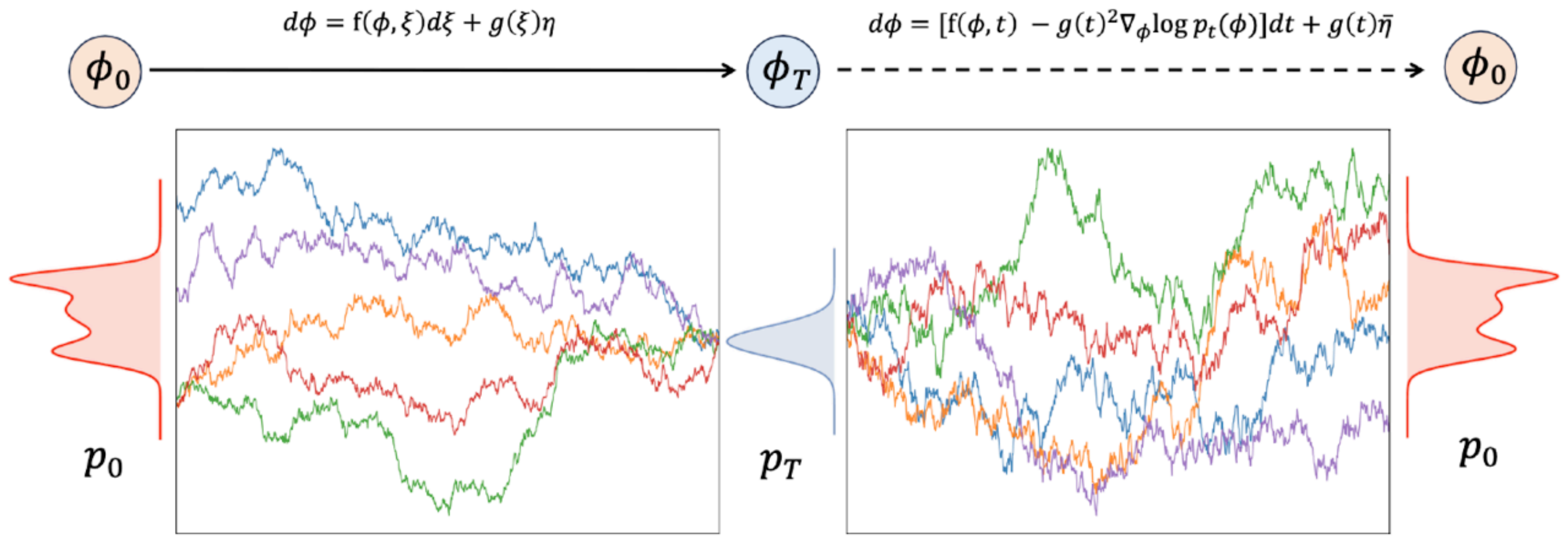


Chinese Phys. Lett. 39, 120502 (2022)

Phys. Rev. D 107, 056001



JHEP 05(2024)060



Thank You !

ML meets Physics, Opportunities and Challenges

Kouzou Sakai @Quanta Magazine

Backups I

1. Building Neutron Star EoS

Phys. Rev. D 107, 083028

Uncertainty estimations

- x : reconstructed EoSs given a sample
- $O(x)$: observables, M, R, P

Recall: Importance Sampling

- x : random variables
- $f(x)$: observables
- n : number of samples
- $p(x)$: original(true) distribution
- $q(x)$: reference distribution

$$E[f(x)] = \int f(x)p(x)dx = \int f(x)\frac{p(x)}{q(x)}q(x)dx$$

Variance $\sigma(O)^2 = \langle \hat{O}^2 \rangle - \bar{O}^2$

Mean $\bar{O} = \langle \hat{O} \rangle = \sum_j^N w^{(j)} O^{(j)}$

Weights

$$w^{(j)} = \frac{\text{Posterior}(\boldsymbol{\theta}_{\text{EoS}}^{(j)} \,|\, \text{data})}{\text{Proposal}(\boldsymbol{\theta}_{\text{EoS}}^{(j)})} \propto \frac{p(\text{data} \,|\, \boldsymbol{\theta}_{\text{EoS}}^{(j)}) \cdot \text{Prior}(\boldsymbol{\theta}_{\text{EoS}}^{(j)})}{p(\boldsymbol{\theta}_{\text{EoS}}^{(j)} \,|\, \text{samples}^{(j)}) \cdot p(\text{samples}^{(j)} \,|\, \text{data}) \cdot \text{Prior}(\text{data})}$$

L2 : $exp(-\lambda||\theta||_2^2) \sim 1, \quad \lambda = 10^{-8}$

$p(\text{data} \,|\, \boldsymbol{\theta}_{\text{EoS}}^{(j)}) \propto \exp(-\chi^2(M_{\boldsymbol{\theta}_{\text{EoS}}^{(j)}}, R_{\boldsymbol{\theta}_{\text{EoS}}^{(j)}}))$

$p(\boldsymbol{\theta}_{\text{EoS}}^{(j)} \,|\, \text{samples}^{(j)}) = 1$

deterministic

$p(\text{samples} \,|\, \text{data}) = \mathcal{N}(M_{\text{obs}}, \Delta M^2)\mathcal{N}(R_{\text{obs}}, \Delta R^2)$

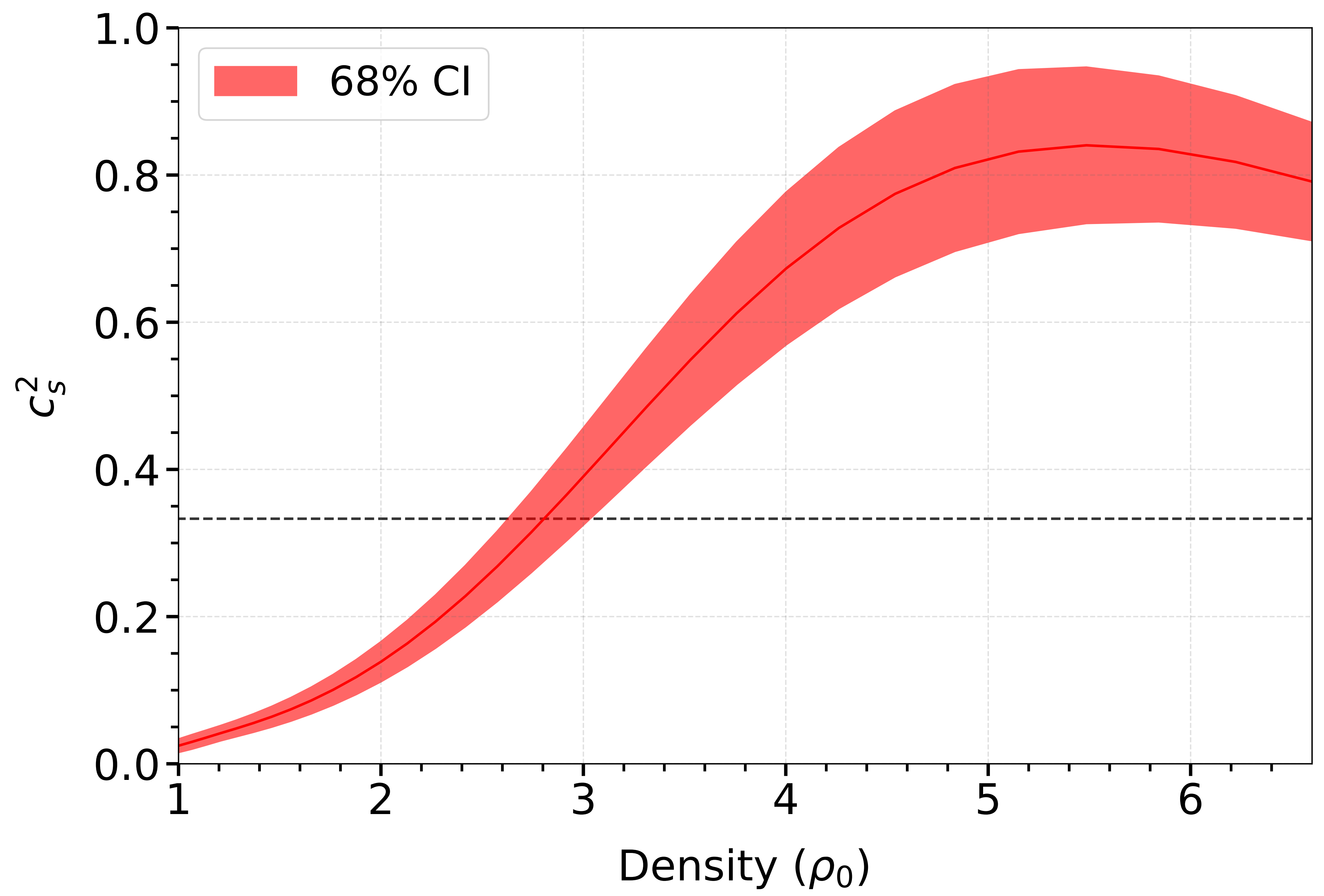
$\tilde{w}^{(j)} = w^{(j)} / \sum_j w^{(j)}$

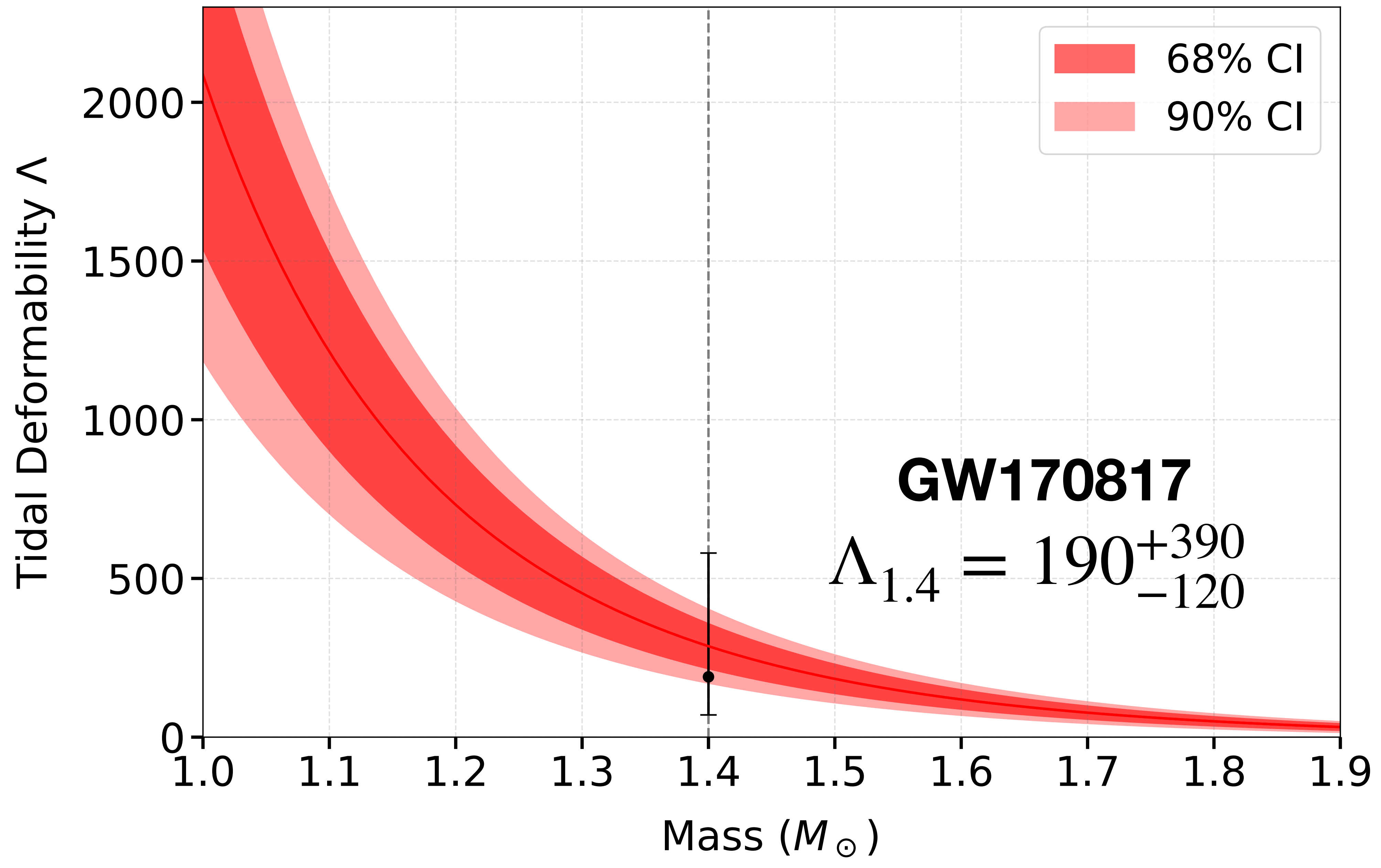
normalization

Backups I

1. Building Neutron Star EoS

Phys. Rev. D 107, 083028





Our results, $\tilde{\Lambda}_{1.4} = 286.47^{+115.9}_{-115.9}$

Backups I

2. Reconstructing Spectral Function

Comput. Phys. Commun. 282, 108547

- In practice, the Euclidean correlations have **finite number of points** and **with finite precision**;
- The ill-posedness of the spectral reconstruction **fundamentally exists even for** *continuous correlation functions(infinite observations)*;
- It's caused by the **numerical inaccuracy** of the correlation measurements (induced high degeneracy in solution space).

$$K_{ij}, i \in N_x, j \in N_\omega, N_x < N_\omega$$

$$\vec{D} \equiv K \vec{\rho}$$

highly rectangular

↑ vectorization

$$D(x) \equiv \int_0^\infty K(x,\omega)\rho(\omega)\mathrm{d}\omega$$

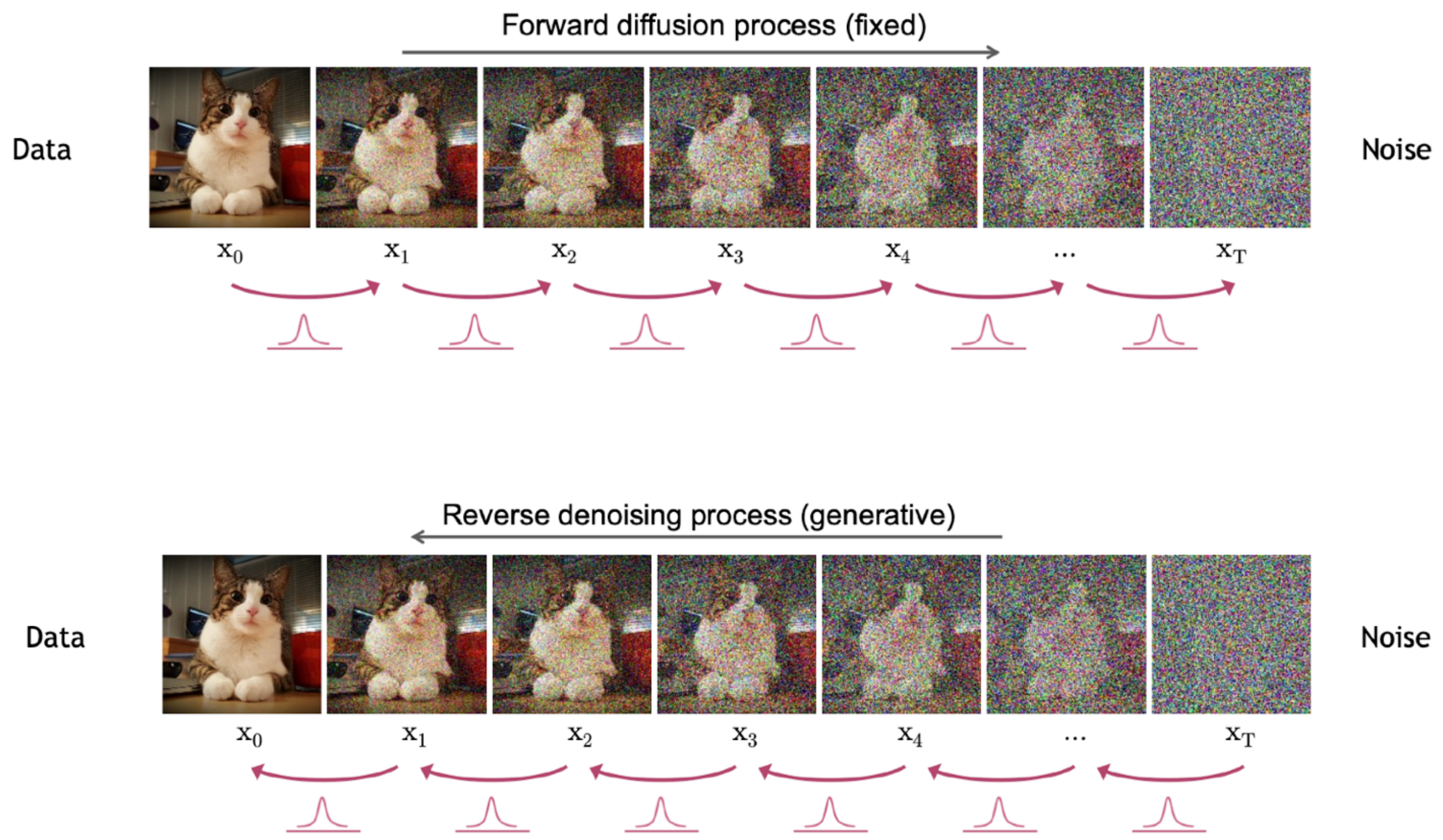
↓ eigenvalue problem

$$\int_0^\infty \psi_s(\omega)\, K(x,\omega)\, \mathrm{d}\omega = \lambda_s \psi_s(x)$$

J. Phys. A: Math. Gen., Vol. 11, No. 9, 1978. Printed in Great Britain.

Backups II: Diffusion Models

- Forward diffusion process **gradually adds noise** to input
- Reverse denoising process learns to **generate data by denoising**
- Train **Probabilistic Models** to learn how to **convert a simple distribution to a target distribution**



Ho et al., Denoising Diffusion Probabilistic Models, NeurIPS 2020

Backups II: Diffusion Models

DMs as SQ

JHEP 05(2024)060

- **Diffusion models(Reverse SDE):**

$$\frac{d\phi}{dt} = -g(t)^2 \nabla_\phi \log p_t(\phi) + g(t)\bar{\eta}$$

  - **Define:** $\tau \equiv T - t (d\tau \equiv -dt)$

$$\frac{d\phi}{d\tau} = g_\tau^2 \nabla_\phi \log q_\tau(\phi) + g_\tau \bar{\eta}$$

$$\phi(\tau_{n+1}) = \phi(\tau_n) + g_\tau^2 \nabla_\phi \log q_{\tau_n}[\phi(\tau_n)]\Delta\tau + g_\tau \sqrt{\Delta\tau}\bar{\eta}(\tau_n)$$

introducing **Noise scale:** $\langle \bar{\eta}^2 \rangle \equiv 2\bar{\alpha}$, **time scale:** $g_\tau^2 \Delta\tau$

- **FP equation**

$$\frac{\partial p_\tau(\phi)}{\partial\tau} = \int d^n x \left\{ g_\tau^2 \bar{\alpha} \frac{\delta}{\delta\phi}\left(\frac{\delta}{\delta\phi} + \frac{1}{\bar{\alpha}} \nabla_\phi S_{\mathbf{DM}}\right)\right\} p_\tau(\phi)$$

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

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

$$p_{\tau=T}(\phi) \to P[\phi, T]$$

$$O(\bar{\alpha}) \sim O(\hbar)$$

The reverse mode of
a well-trained diffusion model at $\tau \to T$ serves as
the stochastic quantization for the input

Backups II: Diffusion Models

DM for Scalar Field

JHEP 05(2024)060

- **Euclidean action** on lattice

$$S_E = \sum_x a^d \Big[\sum_{\mu=1}^{d} \frac{(\phi_0(x+a\hat{\mu}) - \phi_0(x))^2}{a^2} + \frac{m_0^2}{2}\phi_0^2 + \frac{\lambda_0}{4!}\phi_0^4 \Big]$$

- Dimensionless form

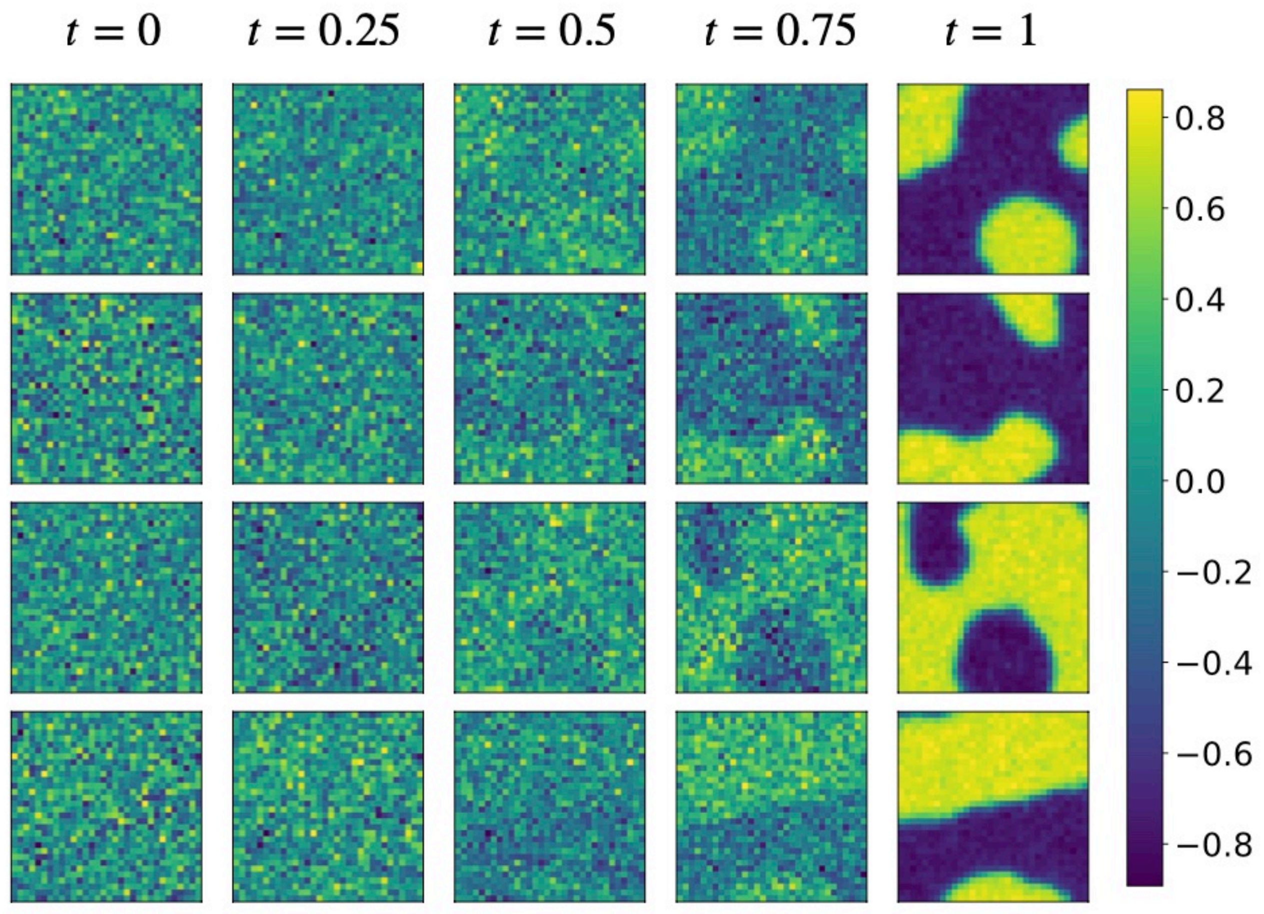
$$S_E = \sum_x \Big[-2\kappa \sum_{\mu=1}^{d} \phi(x)\phi(x+\hat{\mu}) + (1-2\lambda)\phi(x)^2 + \lambda\phi(x)^4 \Big]$$

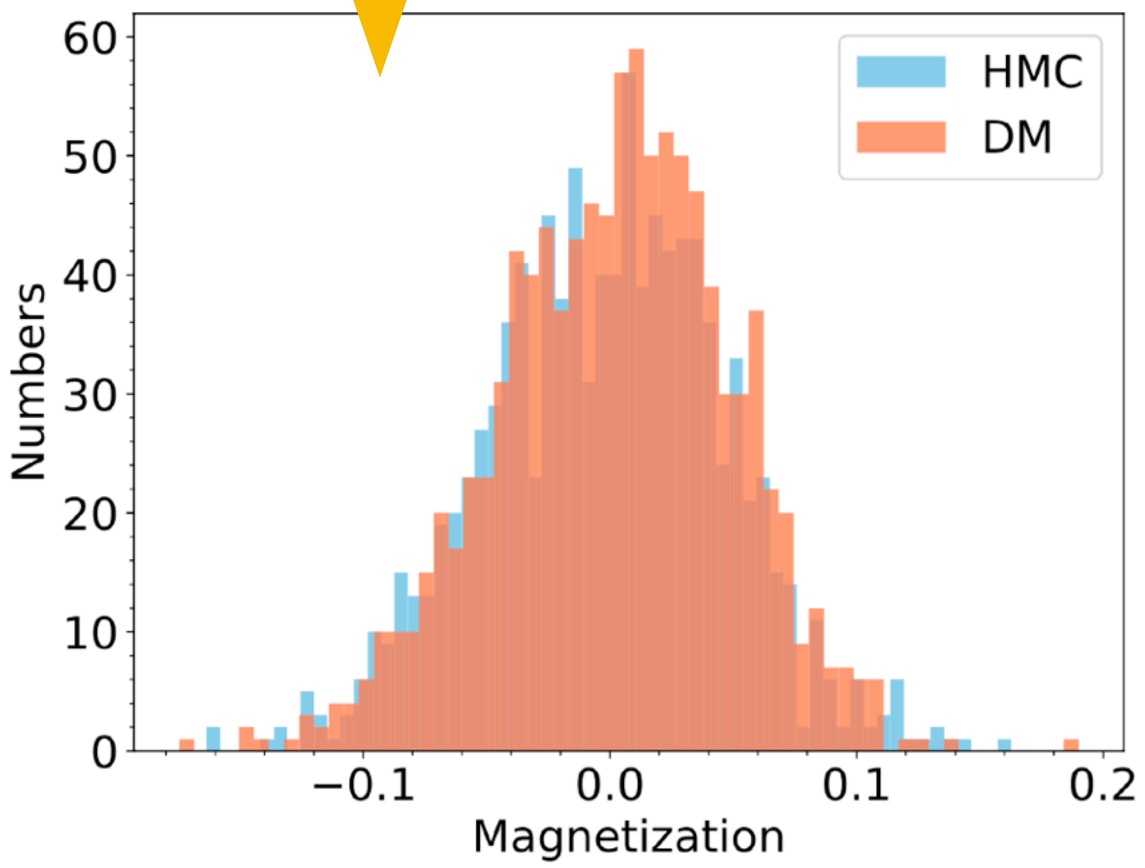
$$a^{\frac{d-2}{2}}\phi_0 = (2\kappa)^{1/2}\phi$$

$$(am_0)^2 = \frac{1-2\lambda}{\kappa} - 2d, \quad a^{-d+4}\lambda_0 = \frac{6\lambda}{\kappa^2}$$

  - Hopping parameter κ, Coupling constant λ

- **Diffusion models**
  - $T = 1.0, \sigma = 25$
- **Data generation**
  - 2-d 32×32 lattice; Hamiltonian Monte Carlo(HMC); 5120 configurations for training.
  - Broken phase: $\kappa = 1.0, \lambda = 0.022$
  - Symmetric phase: $\kappa = 0.21, \lambda = 0.022$





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

Backups III: Autoregressive Networks

$$\max_\theta \prod_{i=1}^{N} p_\theta(\mathbf{s}_i)$$

arXiv:2007.01037

on specific degrees of freedom(*d.o.f.s*)

1. Prepare data-set from observations

$$\mathbf{s} \sim q_{\text{data}}$$

2. Put them into the deep autoregressive network(DAN)

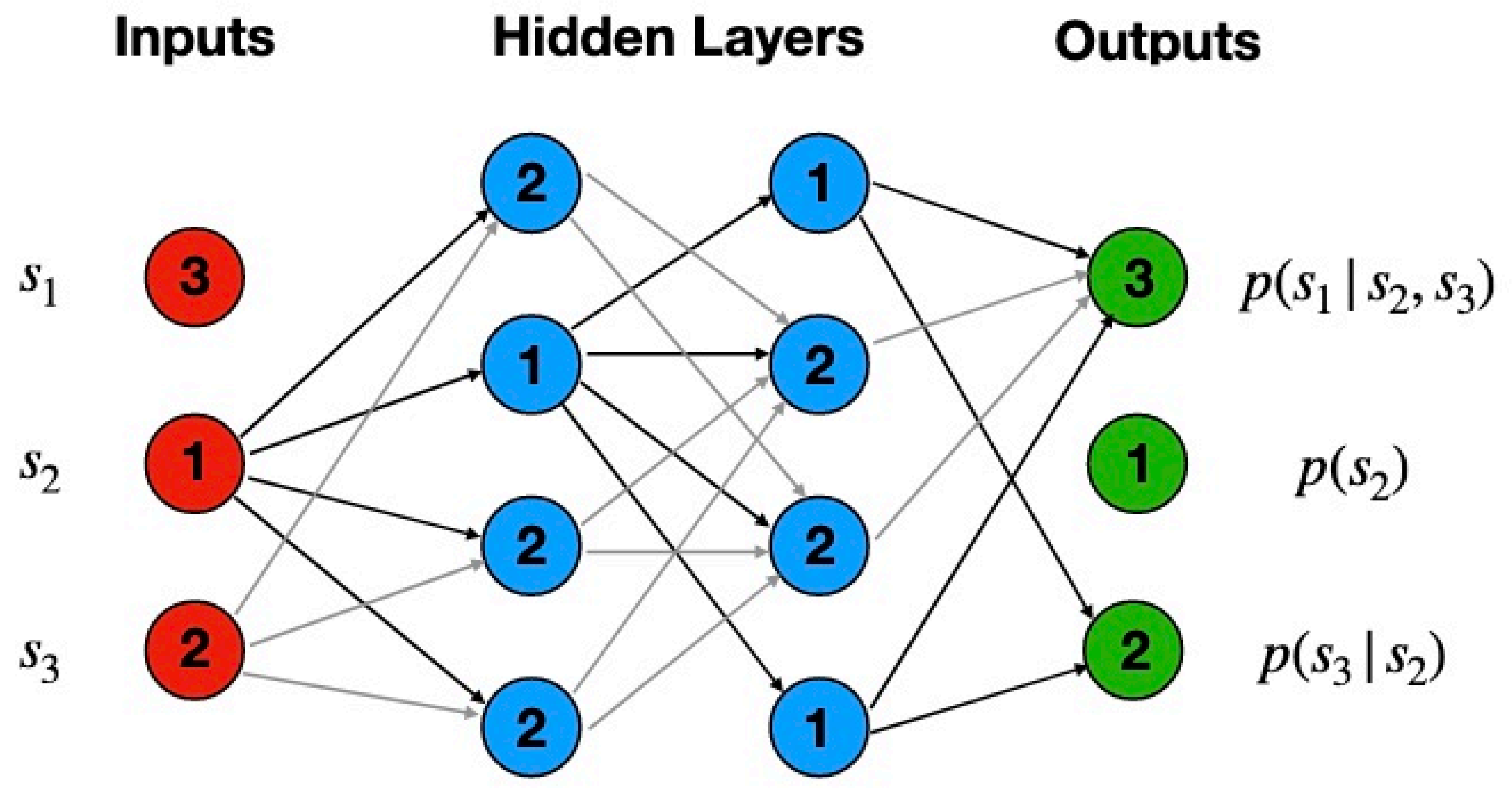
$$p_\theta(\mathbf{s}) = \prod_{i=1}^{N} p(s_i \,|\, s_1, \ldots, s_{i-1}, \theta)$$

3. Minimize the negative log-liklihood(NLL)

$$\mathcal{L} = -\sum_{\mathbf{s}\sim q_{data}} \sum_{d=1}^{N} \log(p(s_d \,|\, \mathbf{s}_{<d}, \theta))$$

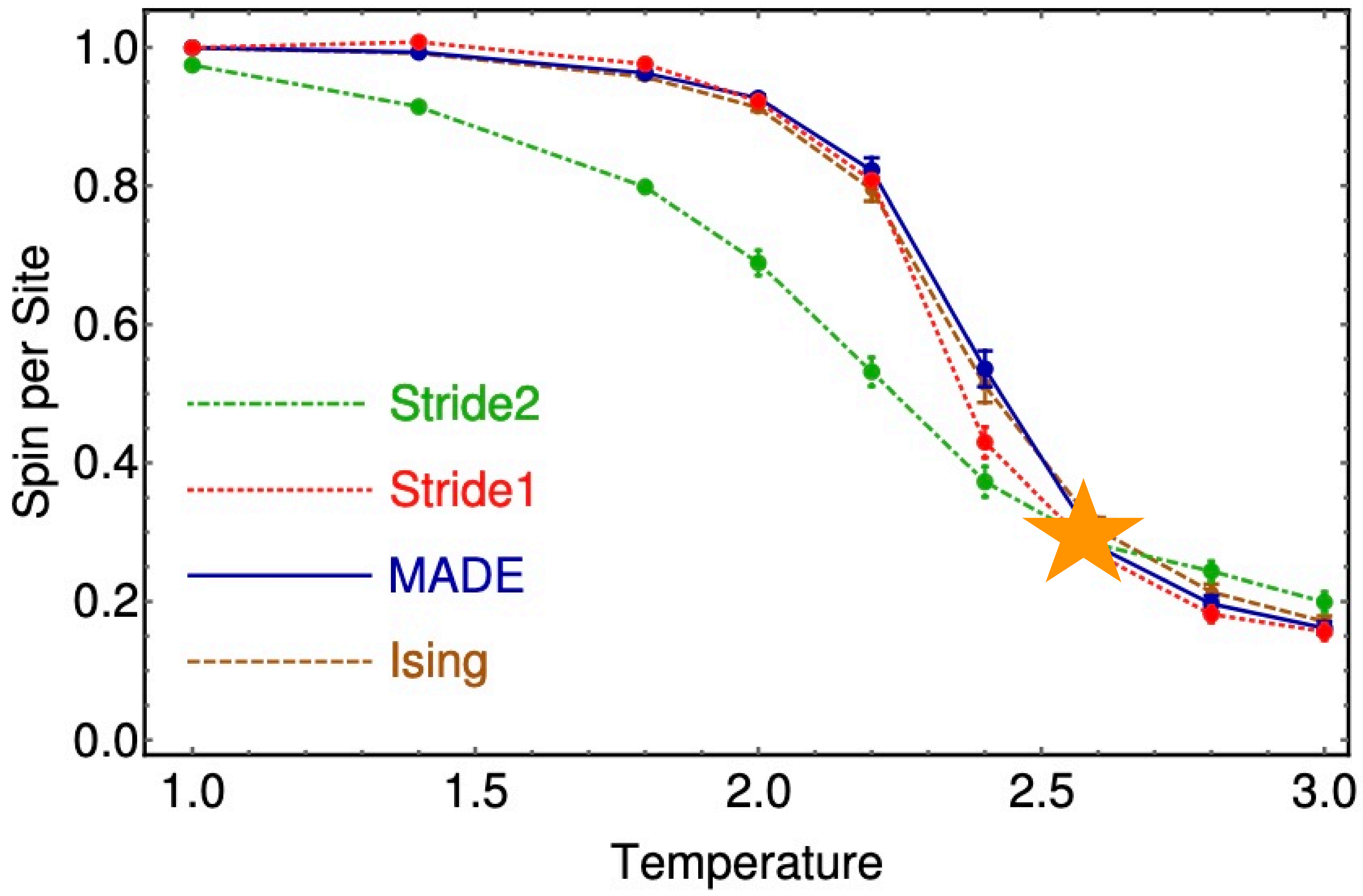
4. Get your DAN represented Hamiltonian

$$H_\theta(\mathbf{s}, T) = -T \ln p_\theta(\mathbf{s})$$

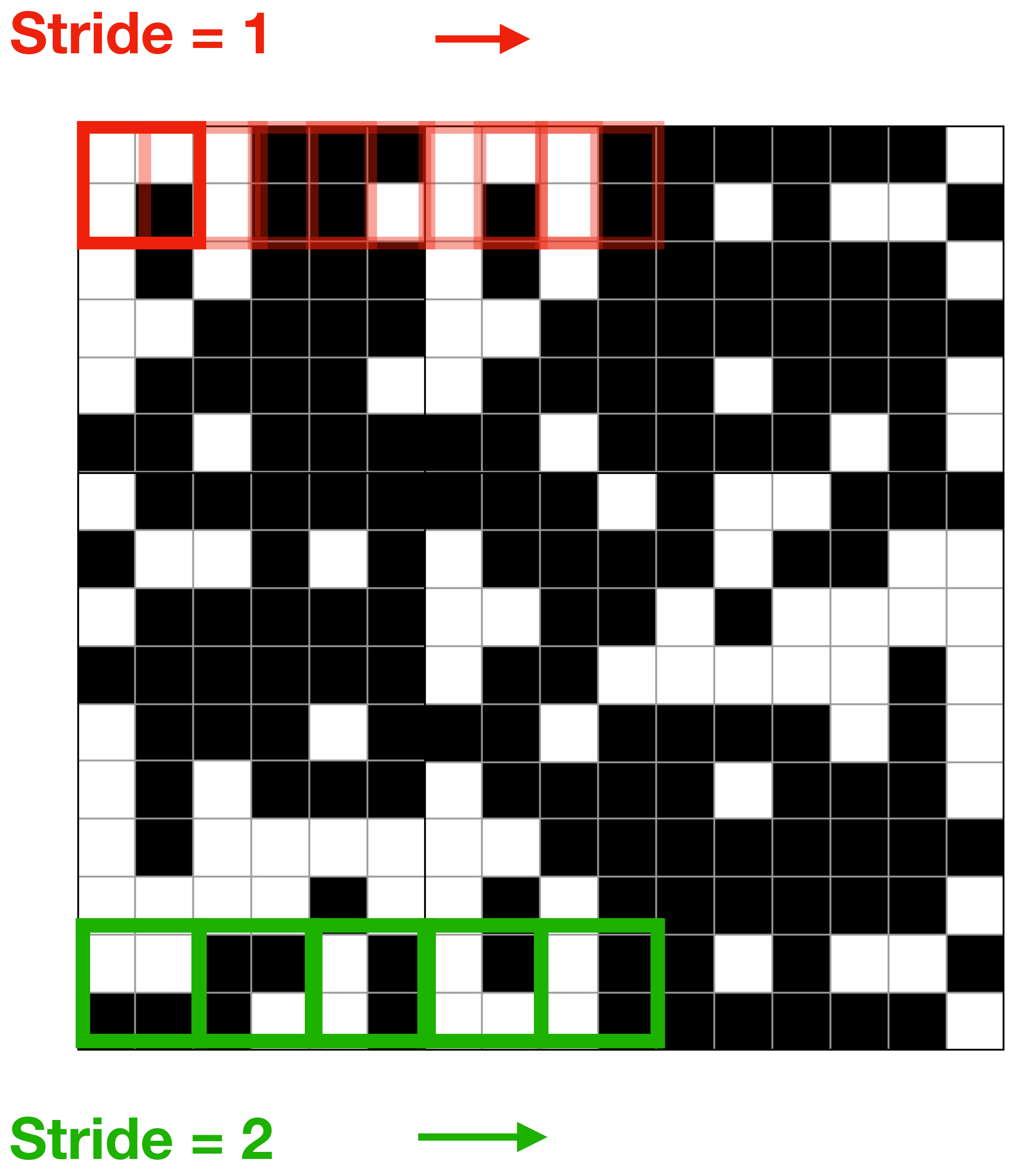


1. 2D Ising Model

arXiv:2007.01037



Masked Autoencoder for Distribution Estimation **(MADE)**



2×2 **spin-block**

Backups III: Finite-Temperature Fields

$$Z = \int D\Phi \exp(-S[\Phi])$$

$$S[\Phi] = \sum \Delta\tau(\Delta x)^3[(\frac{\Delta\Phi}{\Delta\tau})^2 + (\nabla\Phi)^2 + V(\Phi)$$

$$= \sum (\Delta x)^3 \left[\frac{(\Delta\Phi)^2}{\Delta\tau} + \Delta\tau((\nabla\Phi)^2 + V(\Phi))\right]$$

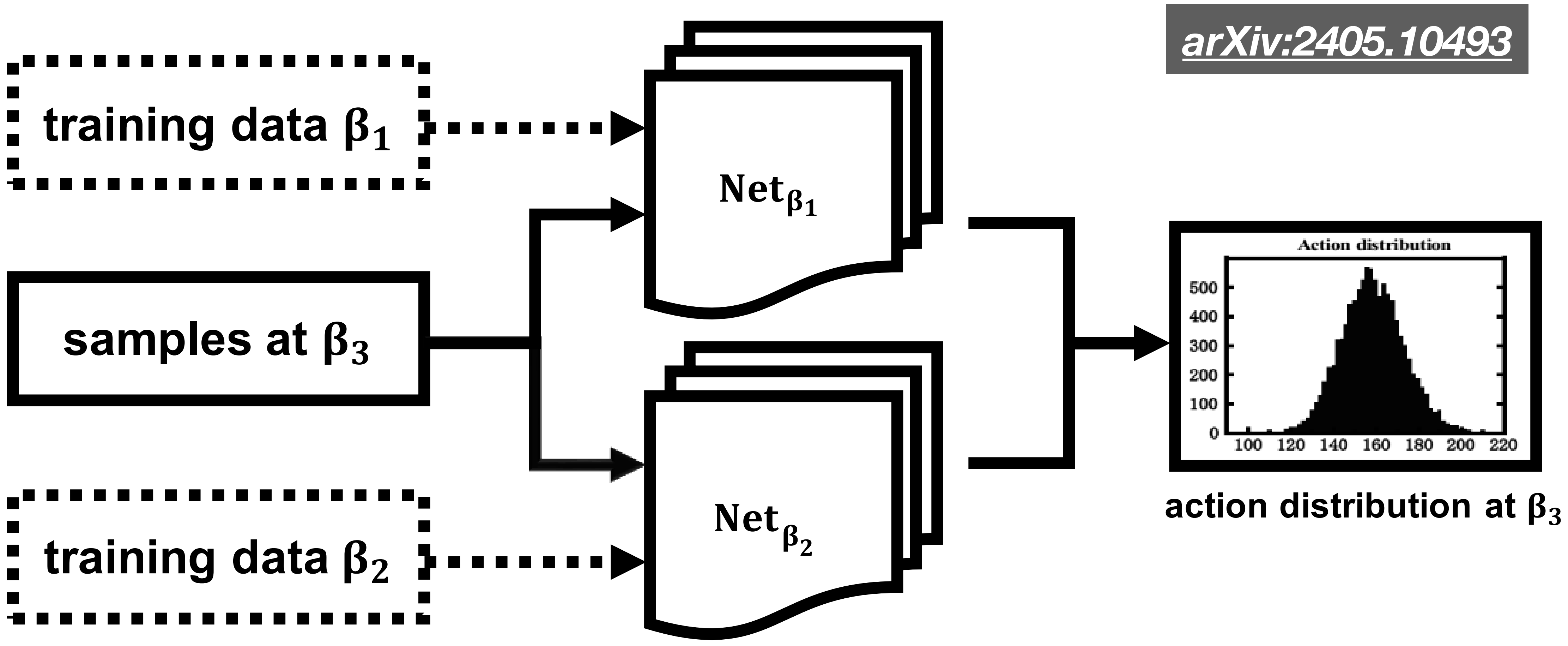
$$= \beta^{-1}K + \beta V$$

$$\Delta\tau = \beta/N_\tau$$

$$K \equiv N_\tau \sum (\Delta x)^3(\Delta\Phi)^2$$

$$V \equiv N_\tau^{-1} \sum (\Delta x)^3[(\nabla\Phi)^2 + V(\Phi)]$$

arXiv:2405.10493



$$S_1[\Phi] = \beta_1^{-1}K[\Phi] + \beta_1 V[\Phi] + C_1$$

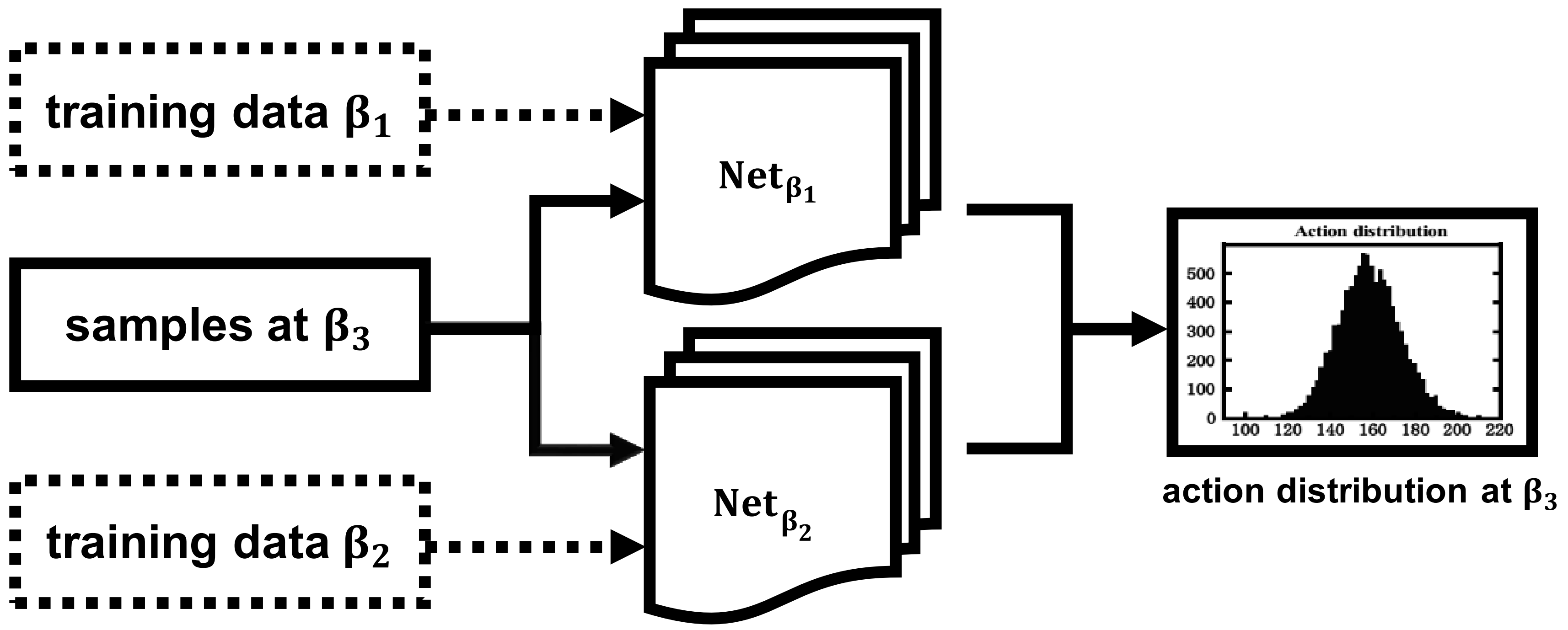
$$S_2[\Phi] = \beta_2^{-1}K[\Phi] + \beta_2 V[\Phi] + C_2,$$

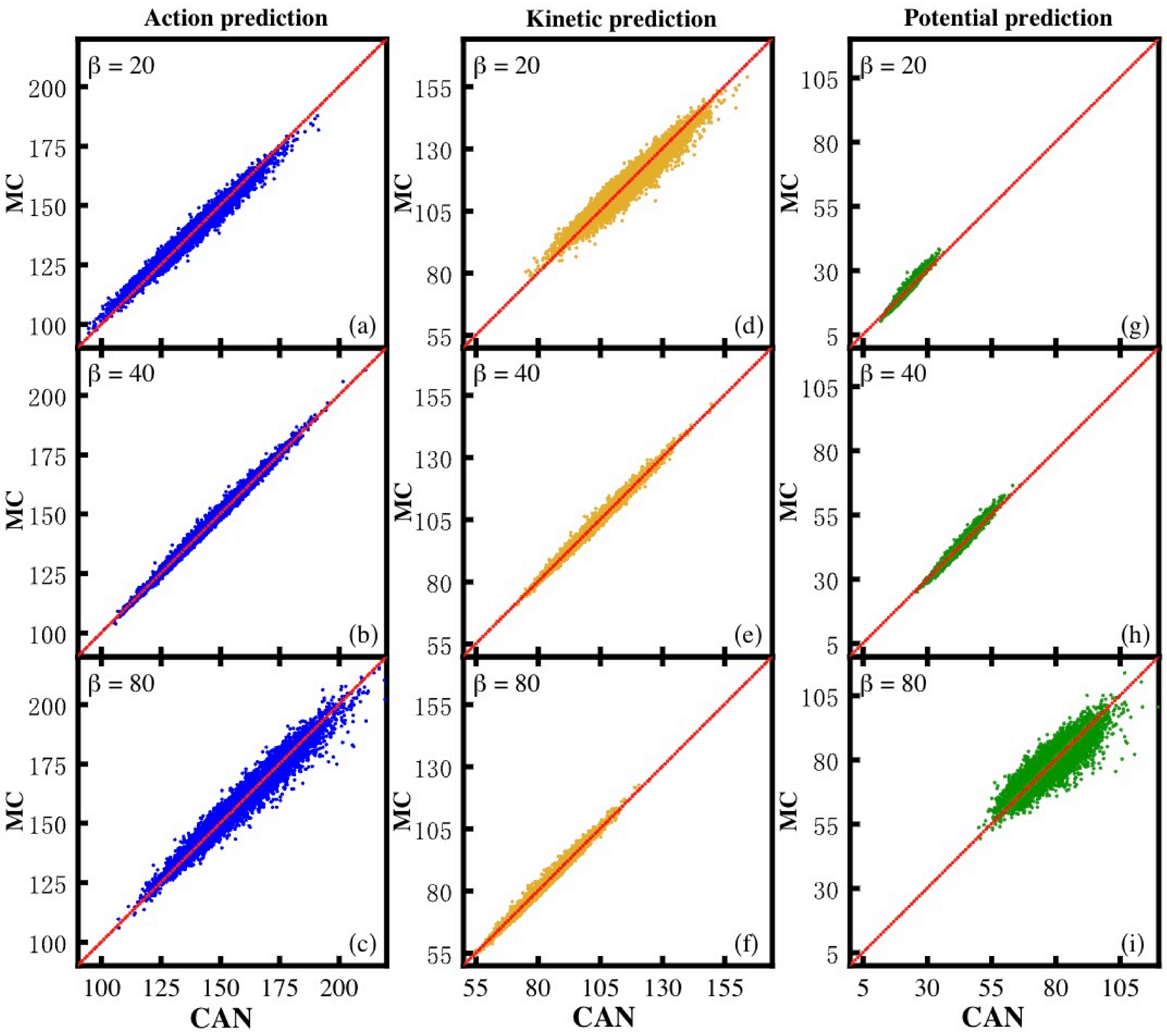
$$S_3[\Phi] = \frac{\beta_1(\beta_3^2 - \beta_2^2)}{\beta_3(\beta_1^2 - \beta_2^2)}S_1 + \frac{\beta_2(\beta_1^2 - \beta_3^2)}{\beta_3(\beta_1^2 - \beta_2^2)}S_2 + C_3$$

$$C_3 = \frac{\beta_1(\beta_2^2 - \beta_3^2)}{\beta_3(\beta_1^2 - \beta_2^2)}C_1 + \frac{\beta_2(\beta_3^2 - \beta_1^2)}{\beta_3(\beta_1^2 - \beta_2^2)}C_2$$

Backups III: Finite-Temperature Fields

2. 0+1 D Quantum Field





arXiv:2405.10493