Fixed point actions from lattice gauge equivariant convolutional neural networks

Andreas Ipp

Institute for Theoretical Physics, TU Wien, Austria

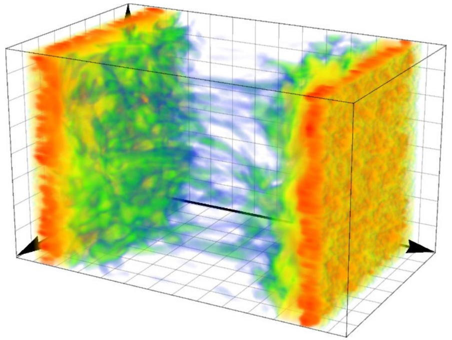
Workshop „Bridging scales: At the crossroads among renormalisation group, multi-scale modelling, and deep learning“

ECT* Trento, April 19, 2024

Der Wissenschaftsfonds.

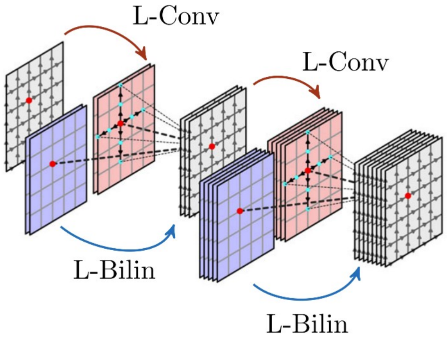
Overview

Glasma simulations



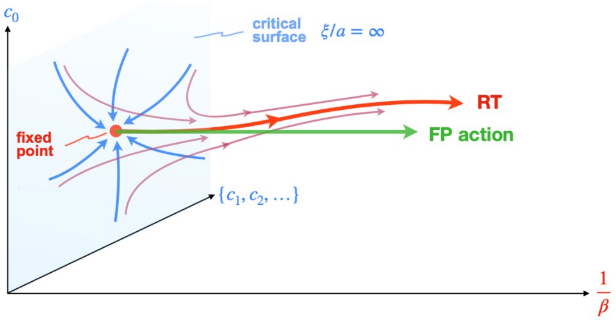
AI, Müller, Phys. Lett. B 771 (2017) 74
Gelfand, AI, Müller, Phys. Rev. D94 (2016) no.1, 014020

L-CNNs



Favoni, AI, Müller, Schuh,
Phys.Rev.Lett. 128 (2022) 032003

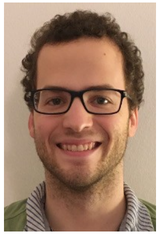
Learning fixed-point actions



Holland, AI, Müller, Wenger
arXiv:2401.06481

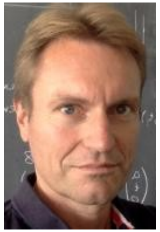
David Müller

Matteo Favoni



Daniel Schuh

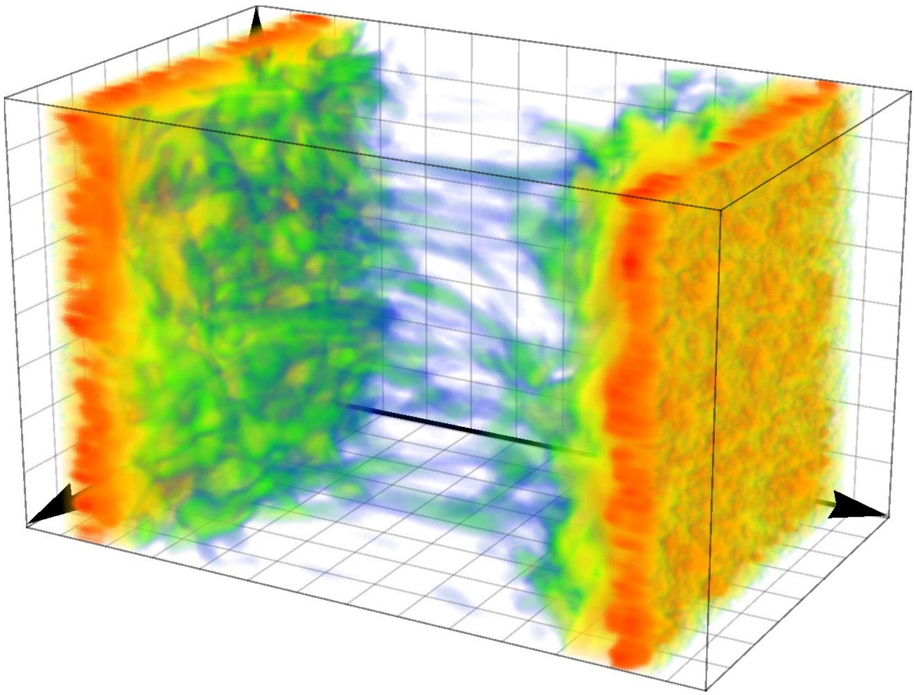
Kieran Holland
(U. Pacific)



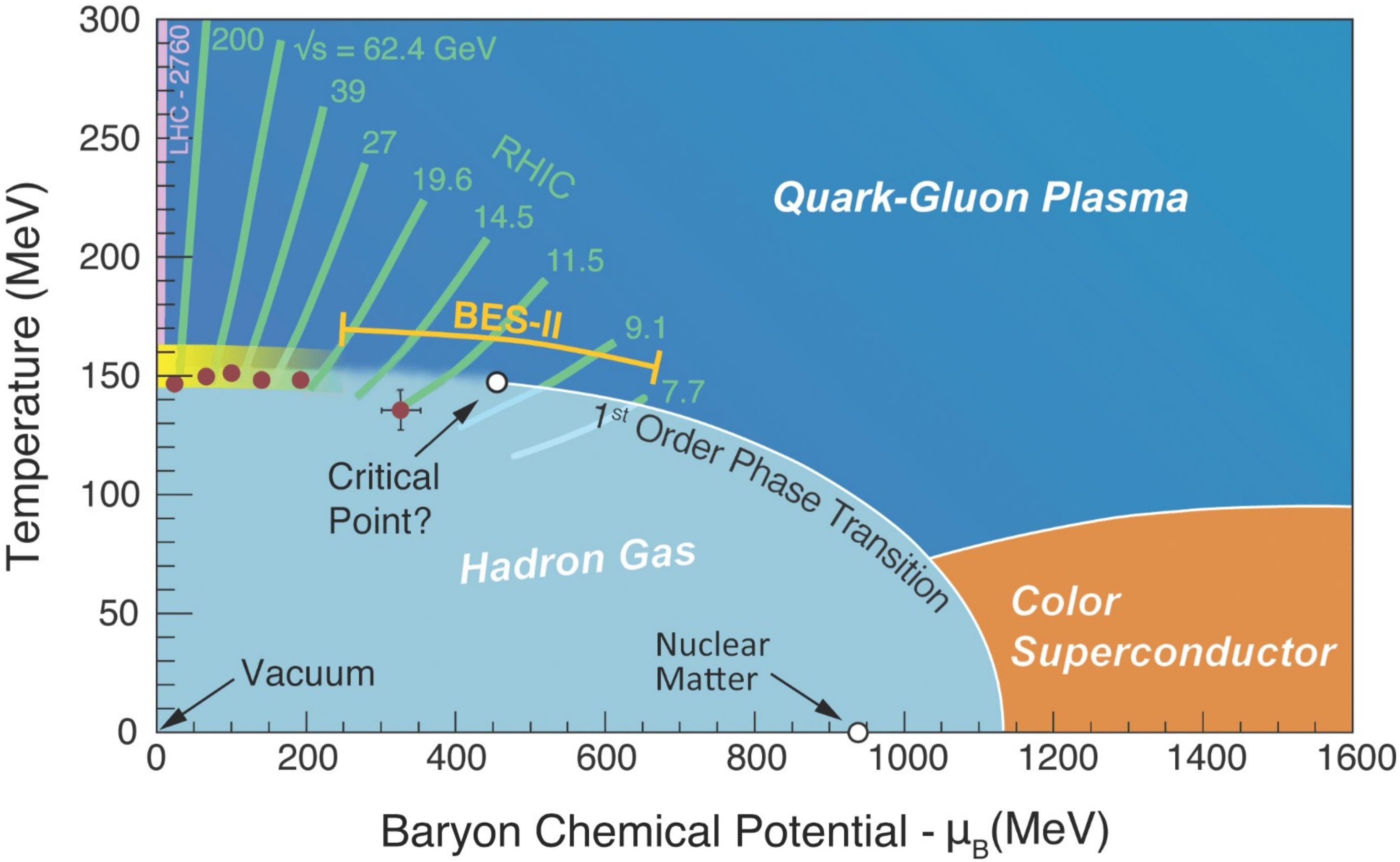
Urs Wenger
(U. Bern)

Open source: https://gitlab.com/openpixi/lge-cnn

Motivation



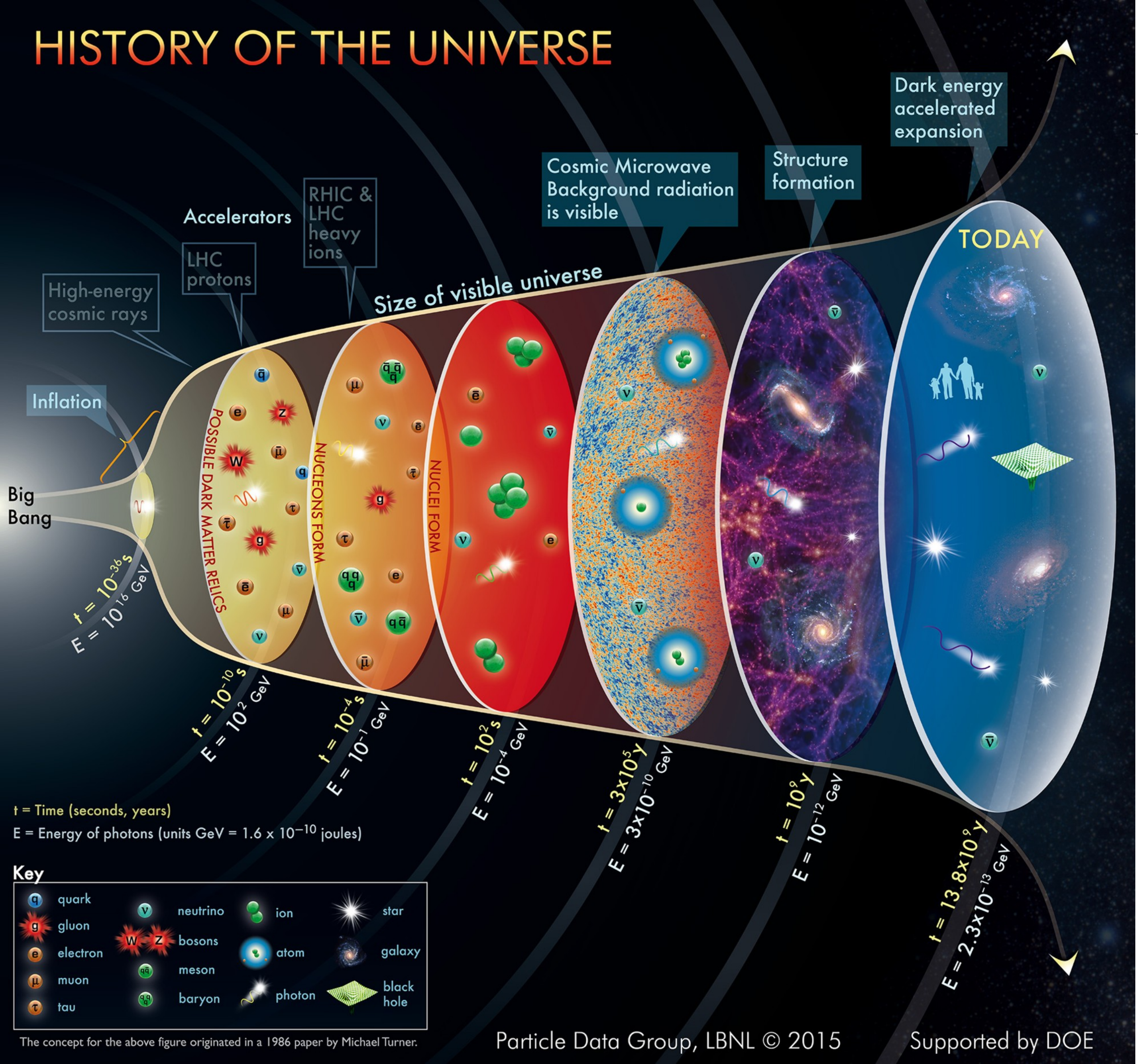
QCD phase diagram



Sun (surface):
6000°C ≈ 0.5 eV

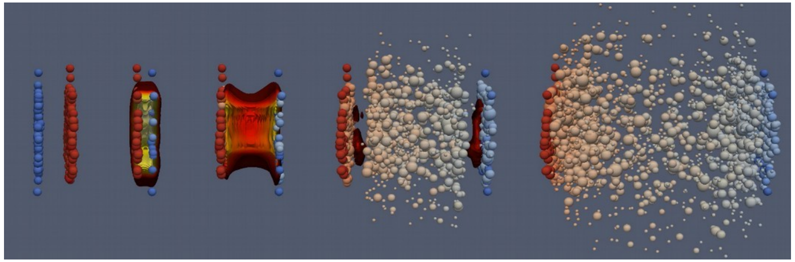
Sun (core):
15 million °C ≈ 1.3 keV

Quark-Gluon Plasma:
1.7×10^{12} °C ≈ 150 MeV

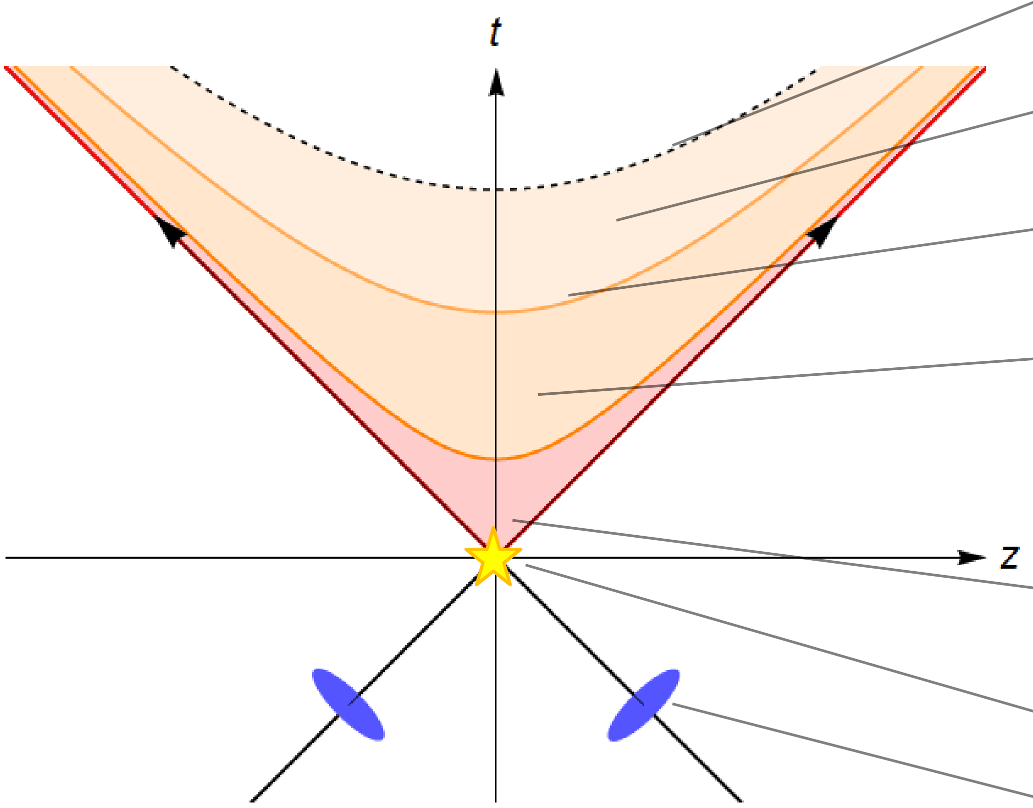


Quark-gluon plasma

- Existed in the early universe
- Produced in heavy ion collisions



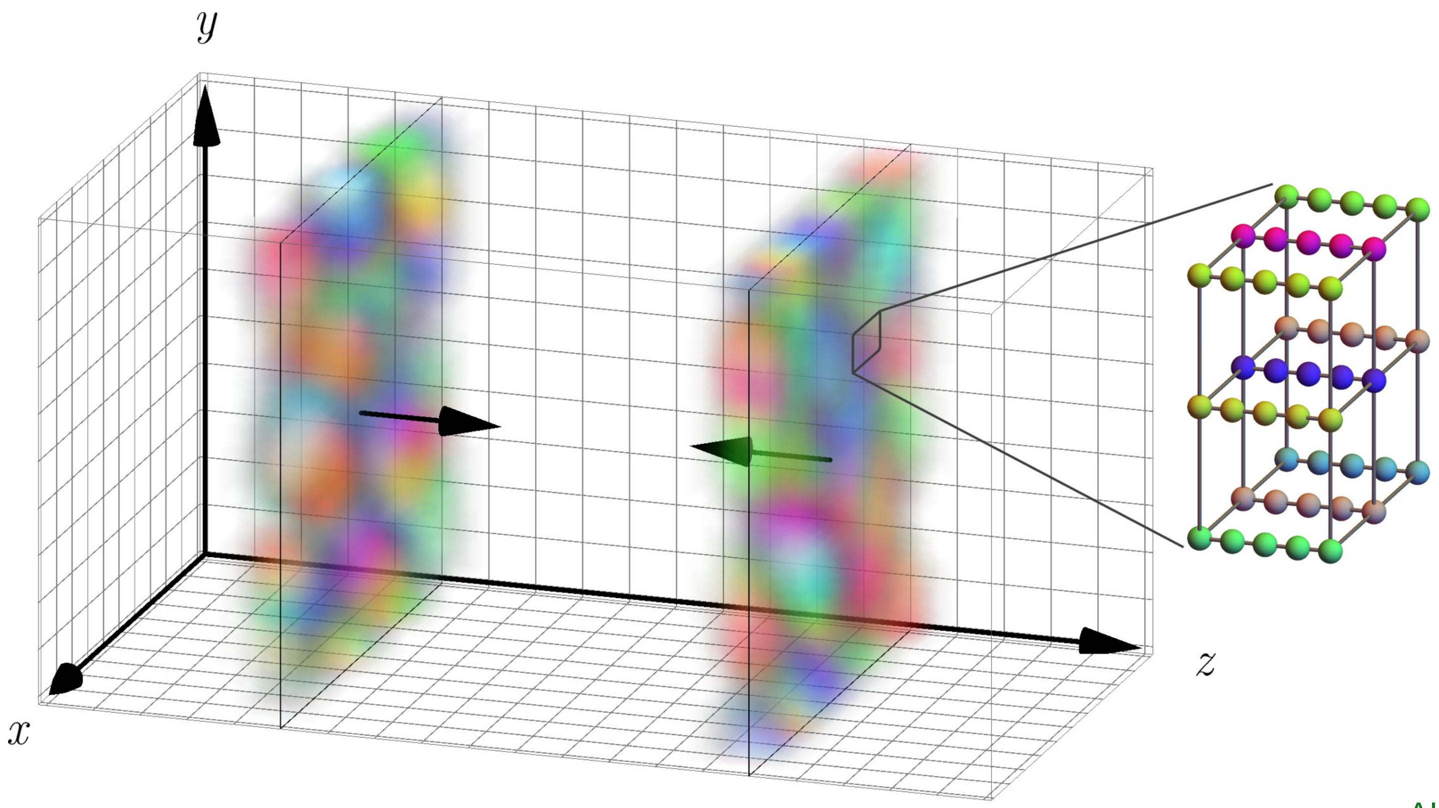
Stages of a heavy-ion collision



- **Freeze-out** (τ ≈ 15 fm/*c*): interactions stop
- **Hadronic gas** (τ ≈ 10 - 15 fm/*c*)**:** hadrons (kinetic transport theory)
- **Hadronization** (τ ≈ 10 fm/*c*): confinement transition → hadron formation
- **QGP** (τ ≈ 1 - 10 fm/*c*)**:** quarks and gluons (relativistic viscous hydrodynamics)
- **Glasma** (τ ≈ 0 - 1 fm/*c*)**:** quasi-classical fields (classical field equations)
- **Collision event**
- **Initial state:** Lorentz-contracted nuclei (color glass condensate)

$$1\,\mathrm{fm}/c \approx 3.3\cdot 10^{-24}\,\mathrm{s} \approx 3.3\,\mathrm{ys}$$

Colored particle-in-cell method



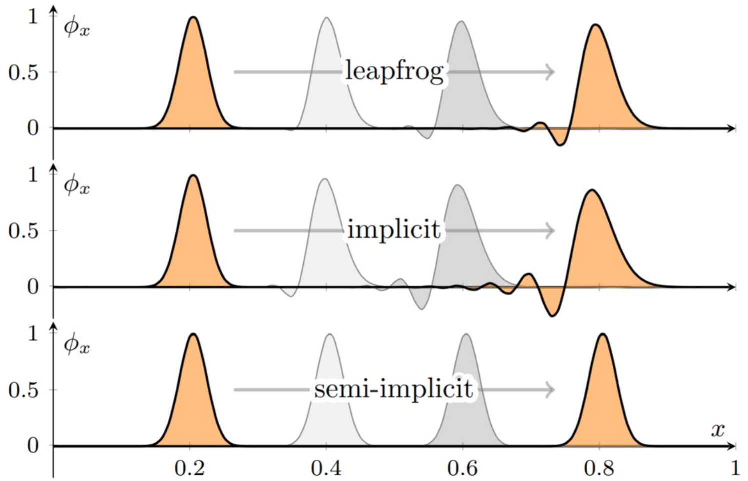
Generalization of the particle-in-cell method from plasma physics for strong interactions.

[A. Dumitru, Y. Nara, M. Strickland: PRD75:025016 (2007)]

Based on real-time lattice gauge theory in a classical regime.

AI, D. Müller, Phys. Lett. B 771 (2017) 74

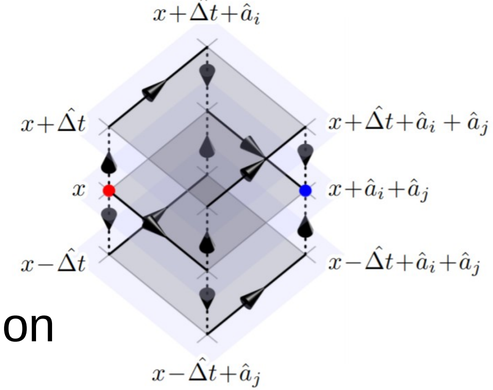
Dispersion-free propagation



Standard **Wilson action**:

$$S[U] = \frac{V}{g^2}\sum_x \left(\sum_i \frac{1}{(a^0 a^i)^2} \mathrm{tr}\left(2 - U_{x,0i} - U^\dagger_{x,0i}\right) - \frac{1}{2}\sum_{i,j} \frac{1}{(a^i a^j)^2} \mathrm{tr}\left(2 - U_{x,ij} - U^\dagger_{x,ij}\right) \right)$$

Variational integrator: Discretized equations of motion from discretized action



Discretized action for the **semi-implicit scheme**:

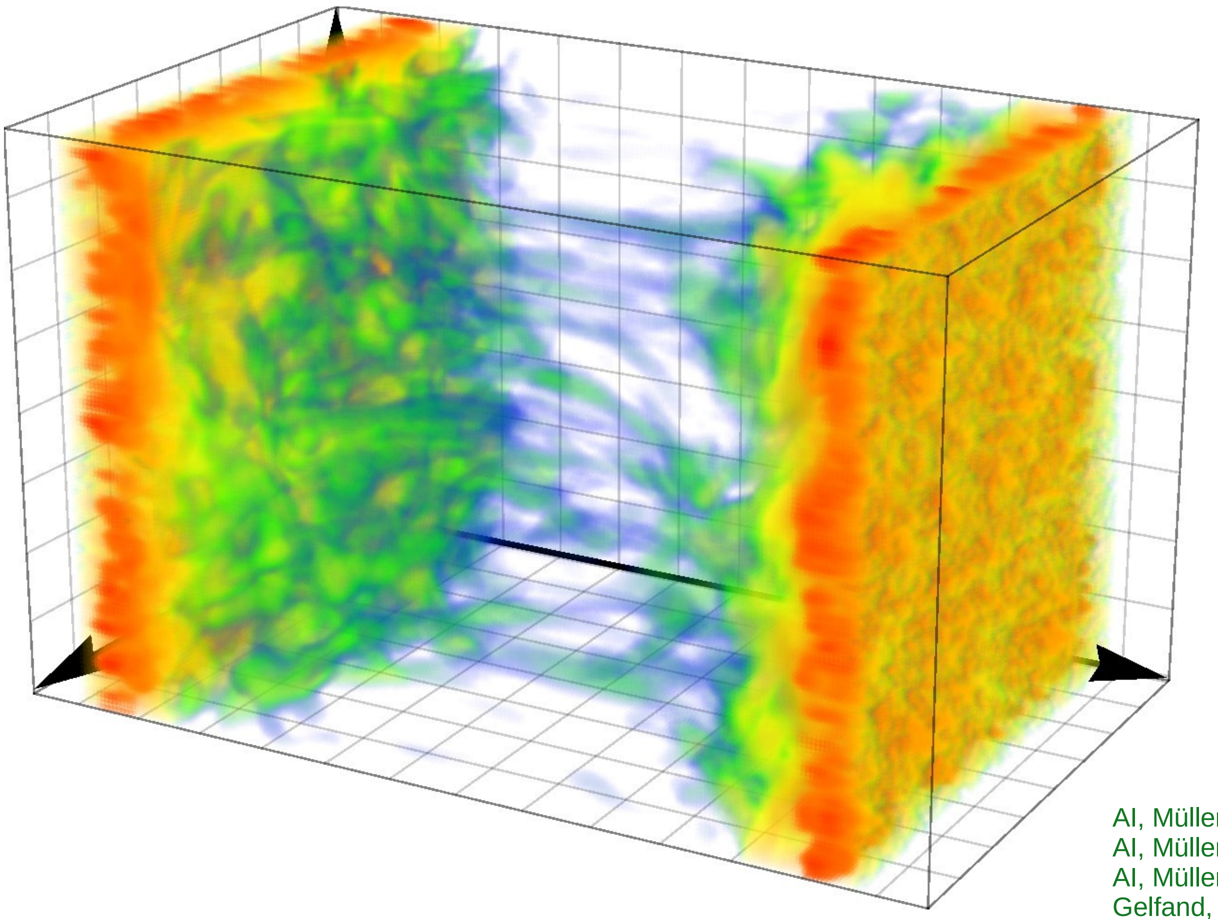
$$S[U] = \frac{V}{g^2}\sum_x \left(\frac{1}{(a^0 a^1)^2} \mathrm{tr}\left(C_{x,01} C^\dagger_{x,01}\right) + \sum_i \frac{1}{(a^0 a^i)^2} \mathrm{tr}\left(C_{x,0i} C^\dagger_{x,0i}\right) - \frac{1}{4}\sum_{i,|j|} \frac{1}{(a^i a^j)^2} \mathrm{tr}\left(C_{x,ij} M^\dagger_{x,ij}\right) - \frac{1}{4}\sum_{|j|} \frac{1}{(a^1 a^j)^2} \mathrm{tr}\left(C_{x,1j} W^\dagger_{x,1j} + \mathrm{h.c.}\right) \right)$$

implicit part   semi-implicit part

with $C_{x,\mu\nu} \equiv U_{x,\mu} U_{x+\mu,\nu} - U_{x,\nu} U_{x+\nu,\mu}$

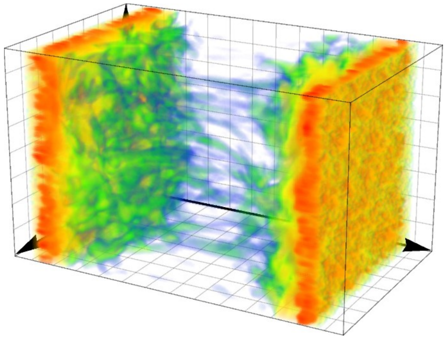
For details see:
AI, D. Müller, Eur.Phys.J. C78 (2018) no.11, 884

Simulations of the collision process



AI, Müller, Eur.Phys.J.A 56 (2020) 9, 243
AI, Müller, Eur.Phys.J. C78 (2018) no.11, 884
AI, Müller, Phys. Lett. B 771 (2017) 74
Gelfand, AI, Müller, Phys. Rev. D94 (2016) no.1, 014020

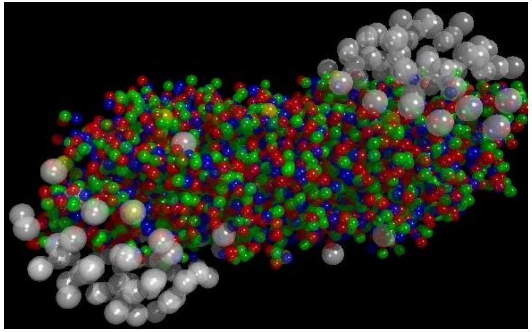
Computational challenges



Simulating small part of nuclei
at RHIC energies:

γ-factor: 100
Lattice: 2048 × 192^2 cells
Gauge group: SU(2)
Color sheets: 1
Simulation box: $(6\ \text{fm})^3$

→ **25 GB** simulation data
→ **192 core hours** on
Vienna Scientific Cluster (VSC-3)

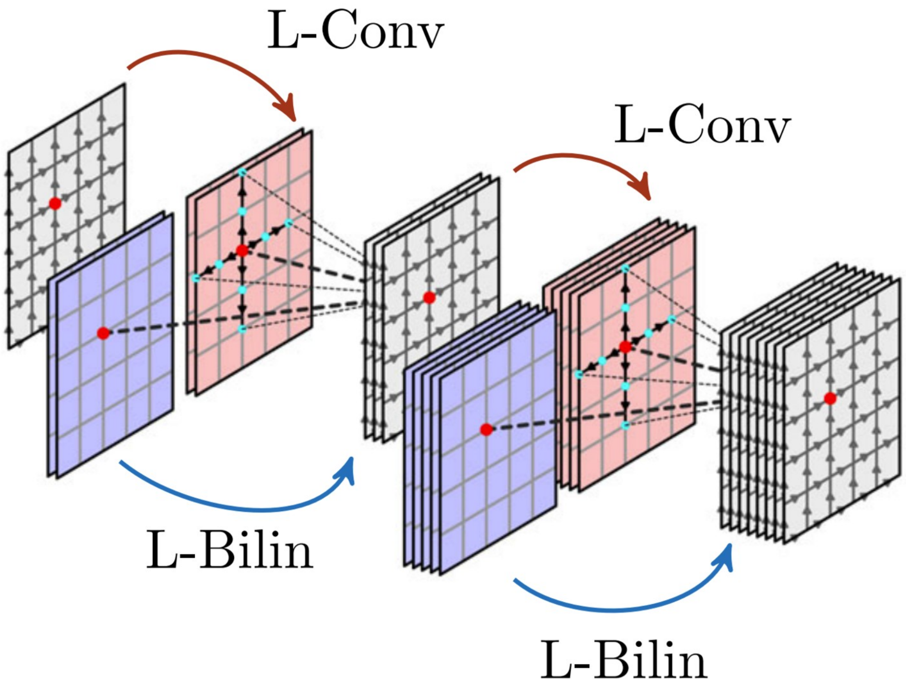


Simulating realistic off-central full size nuclei
at LHC energies:

γ-factor: 2500
Lattice: (25×20480) × 1920^2 cells
Gauge group: SU(3)
Color sheets: 100
Simulation box: $(60\ \text{fm})^3$

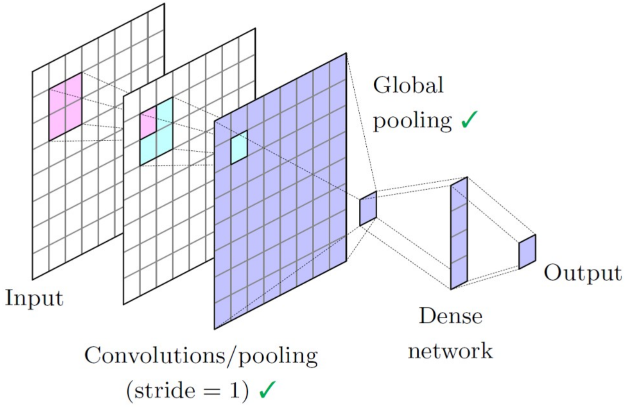
→ **25 PB** simulation data
→ **5 million core years** on VSC-3
(2020: 150 years on VSC-3; but only 130 TB RAM available)
(2023: 55 years on VSC-5; 355 TB RAM available)

Lattice gauge symmetry



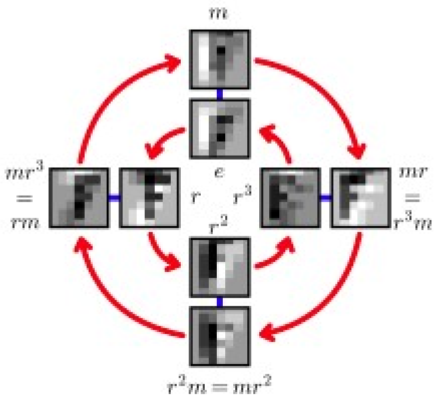
Symmetries on the lattice

Translational symmetry
→ Convolutional neural networks (CNNs)



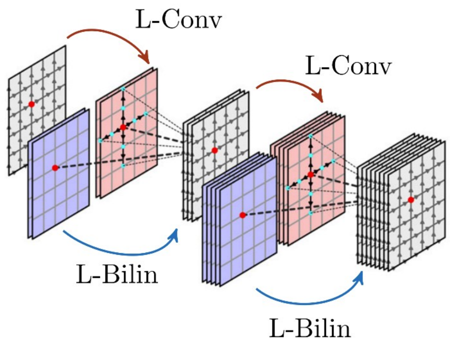
Bulusu, Favoni, AI, Müller, Schuh,
Phys. Rev. D 104 (2021) 074504

Rotation, mirror symmetry
→ Group equivariant CNNs (G-CNNs)



Cohen, Welling, ICML 2016

Lattice gauge symmetry
→ Lattice gauge equivariant CNNs (L-CNNs)



Favoni, AI, Müller, Schuh,
Phys.Rev.Lett. 128 (2022) 032003

Yang-Mills action vs. Wilson action

Continuum formulation

Yang-Mills action

$$S_G[A] = \frac{1}{2\,g^2}\int \mathrm{d}^4x \; \mathrm{tr}\left[F_{\mu\nu}(x)F_{\mu\nu}(x)\right]$$

Field strength tensor

$$F_{\mu\nu}(x) = -\mathrm{i}[D_\mu(x), D_\nu(x)]$$
$$= \partial_\mu A_\nu(x) - \partial_\nu A_\mu(x) + \mathrm{i}[A_\mu(x), A_\nu(x)]$$

Covariant derivative

$$D_\mu(x) = \partial_\mu + \mathrm{i}\,A_\mu(x)$$

SU(3) gauge fields

$$A_\mu(x) = \sum_{i=1}^{8} A_\mu^{(i)}(x)\,T_i$$

Gauge transformation

$$A_\mu(x) \to A'_\mu(x) = \Omega(x)A_\mu(x)\Omega(x)^\dagger + \mathrm{i}\left(\partial_\mu \Omega(x)\right)\Omega(x)^\dagger$$

Discrete formulation

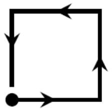
Wilson action

$$S_W[U] = \frac{2}{g^2}\sum_{x\in\Lambda}\sum_{\mu<\nu}\mathrm{Tr}\left[\mathbb{1} - U_{x,\mu\nu}\right]$$

Wilson (1974)

Plaquette

$$U_{x,\mu\nu} = U_{x,\mu}U_{x+\mu,\nu}U^\dagger_{x+\nu,\mu}U^\dagger_{x,\nu} =$$



Link variable

$$U_\mu(n) = \exp\left(\mathrm{i}\,aA_\mu(n)\right)$$

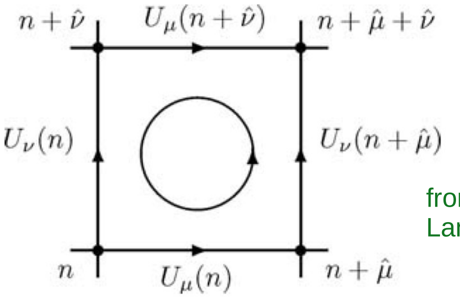
Gauge transformation

$$U_\mu(n) \;\to\; U'_\mu(n) = \Omega(n)\,U_\mu(n)\,\Omega(n+\hat{\mu})^\dagger$$

Taylor expansion in small lattice spacing reproduces continuum action:

$$U_{\mu\nu}(n) = \exp\left(\mathrm{i}\,a^2F_{\mu\nu}(n) + \mathcal{O}(a^3)\right)$$

$$S_G[U] = \frac{2}{g^2}\sum_{n\in\Lambda}\sum_{\mu<\nu}\mathrm{Re}\;\mathrm{tr}\left[\mathbb{1} - U_{\mu\nu}(n)\right] = \frac{a^4}{2\,g^2}\sum_{n\in\Lambda}\sum_{\mu,\nu}\mathrm{tr}[F_{\mu\nu}(n)^2] + \mathcal{O}(a^2)$$



from Gattringer, Lang (2010)

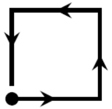
Wilson loops

Wilson action

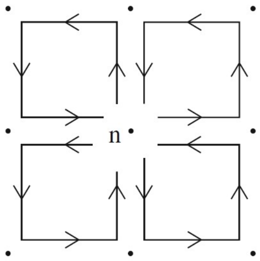
$$S_W[U] = \frac{2}{g^2} \sum_{x\in\Lambda} \sum_{\mu<\nu} \mathrm{Tr}\left[\mathbb{1} - U_{x,\mu\nu}\right]$$

Plaquette

$$U_{x,\mu\nu} = U_{x,\mu} U_{x+\mu,\nu} U^\dagger_{x+\nu,\mu} U^\dagger_{x,\nu} =$$

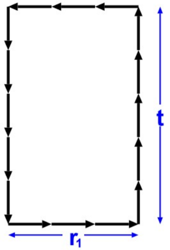


Symanzik improved clover action



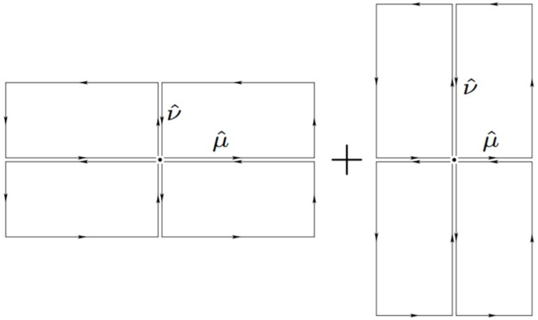
from: Gattringer, Lang (2010)

Potential of static quark pair



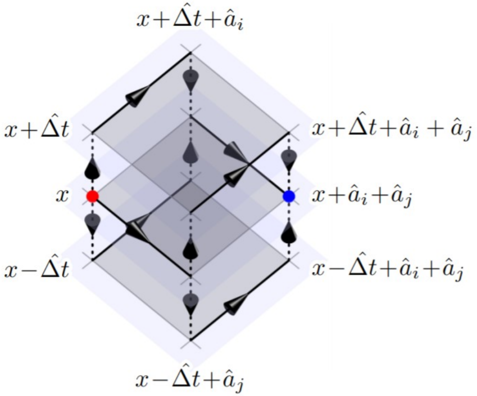
from: Bali, Phys.Rept. 343:1 (2001)

Improved topological charge



from: Alexandrou et al., Eur.Phys.J.C 80 (2020) 5, 424

Improved real-time lattice actions



AI, Müller, Eur.Phys.J. C78 (2018) no.11, 884

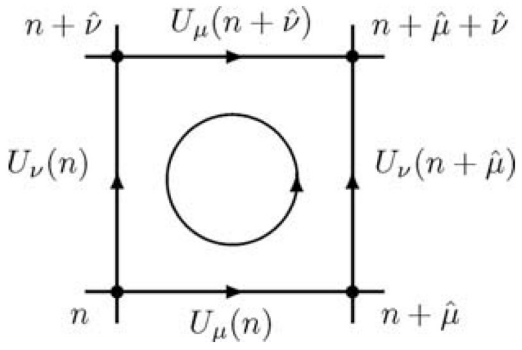
L-CNN data

Combine lattice links *U*
and locally transforming objects *W*

$$\text{tuple } (\mathcal{U}, \mathcal{W})$$

$$\mathcal{U} = \{U_{x,\mu}\} \text{ SU}(N_c) \text{ matrices}$$

$$\mathcal{W} = \{W_{x,i}\} \text{ with } W_{x,i} \in \mathbb{C}^{N_c \times N_c}$$



from: Gattringer, Lang (2010)

Gauge transformation

$$T_\Omega U_{x,\mu} = \Omega_x U_{x,\mu} \Omega^\dagger_{x+\mu}$$

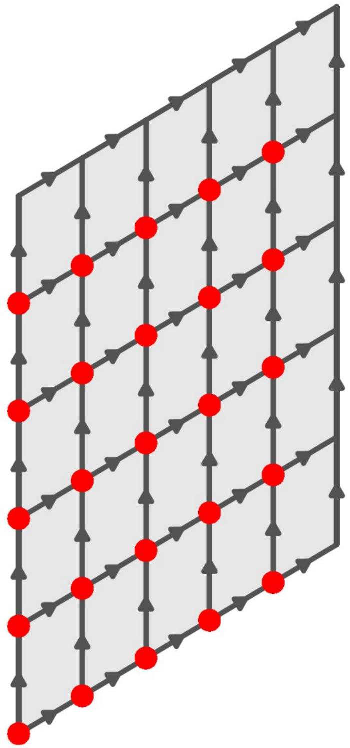
$$T_\Omega W_{x,i} = \Omega_x W_{x,i} \Omega^\dagger_x$$

Gauge equivariant (gauge covariant) function

$$f(T_\Omega \mathcal{U}, T_\Omega \mathcal{W}) = T'_\Omega f(\mathcal{U}, \mathcal{W})$$

Gauge invariant function

$$f(T_\Omega \mathcal{U}, T_\Omega \mathcal{W}) = f(\mathcal{U}, \mathcal{W})$$

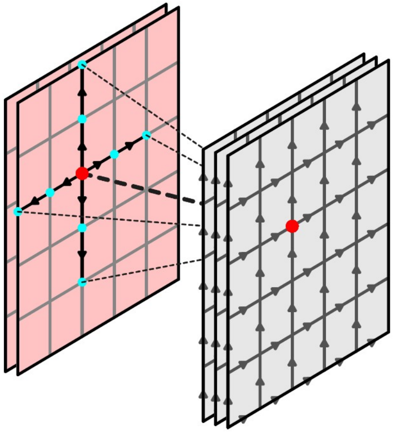


$$\mathcal{U} = \{U_{\mathbf{x},\mu}\}$$

$$\mathcal{W} = \{W_{\mathbf{x},\mu\nu}\}$$

Lattice gauge equivariant layers

Convolution (L-Conv)

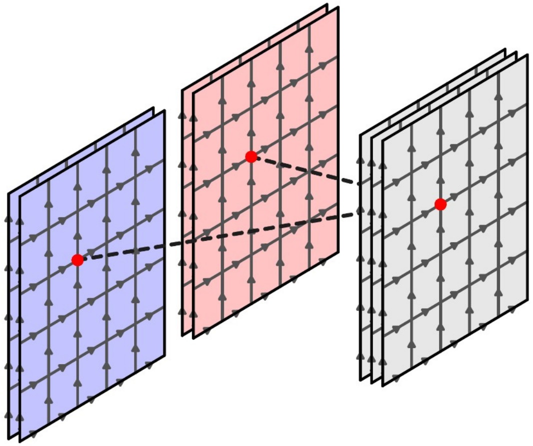


Convolution wish shared weights and proper parallel transport along coordinate axes

$$(\mathcal{U},\mathcal{W}) \to (\mathcal{U},\mathcal{W}')$$

$$W'_{\mathbf{x},i} = \sum_{j,\mu,k} \omega_{i,j,\mu,k} U_{\mathbf{x},k\cdot\mu} W_{\mathbf{x}+k\cdot\mu,j} U^\dagger_{\mathbf{x},k\cdot\mu}$$

Bilinear layer (L-Bilin)

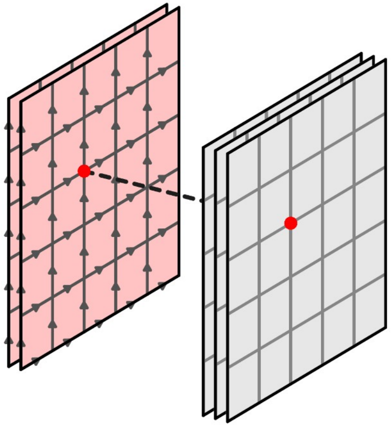


Multiply *W* at each lattice point

$$(\mathcal{U},\mathcal{W}) \times (\mathcal{U},\mathcal{W}') \to (\mathcal{U},\mathcal{W}'')$$

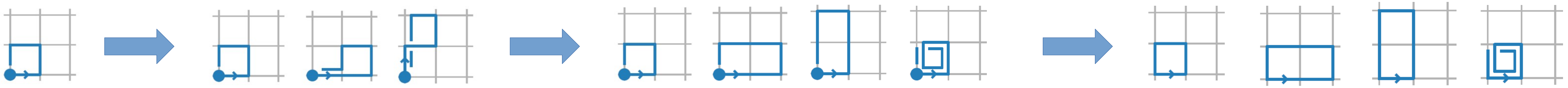
$$W''_{\mathbf{x},i} = \sum_{j,k} \alpha_{ijk} W_{\mathbf{x},j} W'_{\mathbf{x},k}$$

Trace layer

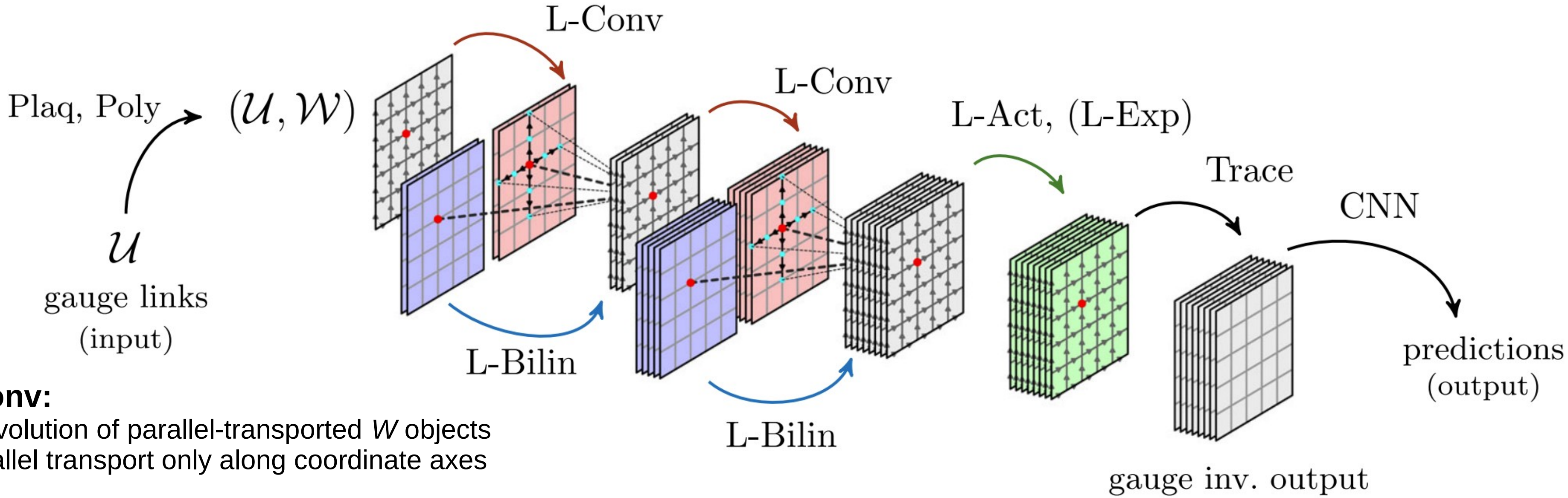


Generate gauge invariant output

$$w_{\mathbf{x},i} = \mathrm{Tr}\, W_{\mathbf{x},i} \in \mathbb{C}$$



Generic L-CNN



L-Conv:
* convolution of parallel-transported *W* objects
* parallel transport only along coordinate axes

L-Bilin:
* bilinear layer, product of locally transforming objects

L-Act:
* activation functions multiply *W* objects by scalar, gauge-invariant functions

L-Exp:
* update link variables using exponential map

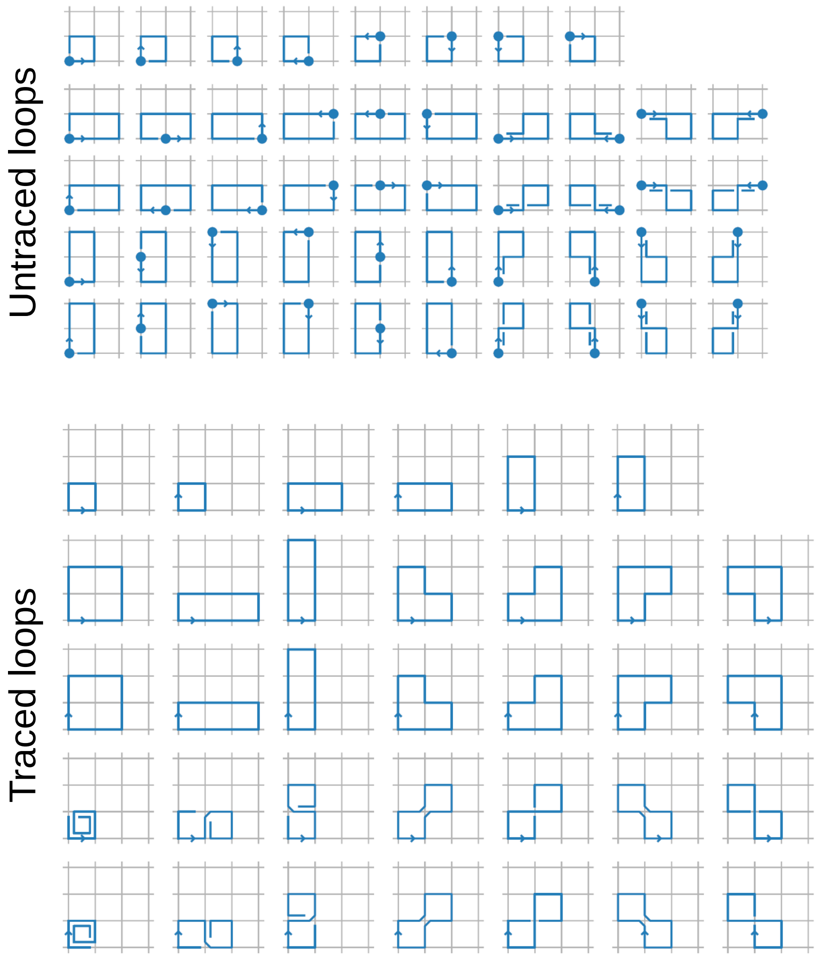
Trace:
* calculate gauge invariant trace

Plaq:
* generate all possible plaquettes

Poly:
* generate all possible Polyakov loops

Favoni, AI, Müller, Schuh,
Phys.Rev.Lett. 128 (2022) 032003

L-CNNs generate Wilson loops



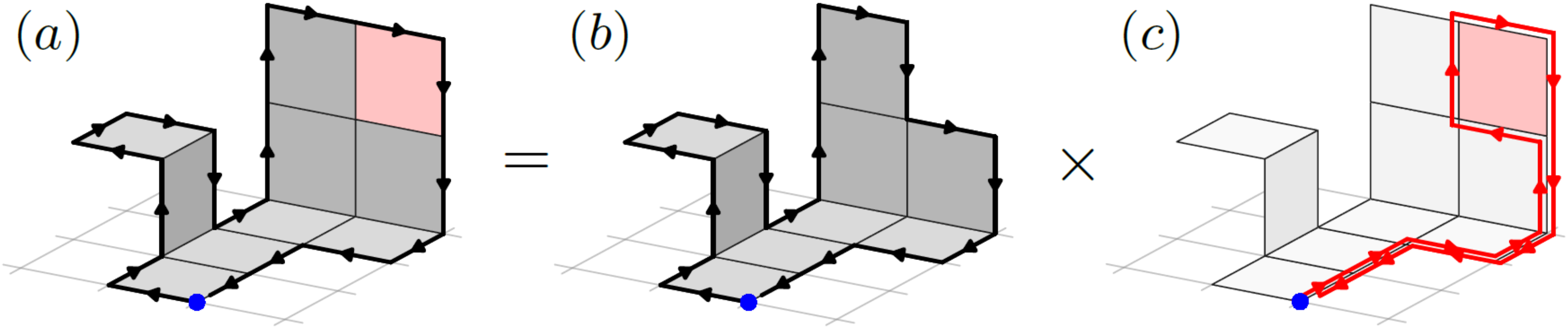
Number of traced Wilson loops covered by L-CNN architectures of various sizes in 1+1 D

Length	Max	$W^{(1\times1)}$	$W^{(1\times2)}$			$W^{(2\times2)}$		
		S	S	M	L	S	M	L
0	1	1	1	1	1	1	1	1
2	0	0	0	0	0	0	0	0
4	2	2	2	2	2	2	2	2
6	4		4	4	4	4	4	4
8	28		4	4	4	22	22	22
10	152			8	8	48	76	76
12	1,010				8	92	204	220
14	6,772					120	412	532
16	47,646					100	712	1,080
18	343,168					136	928	1,896
20	2,529,890					32	1,056	2,620
22	18,982,172					64	768	3,152
$\geq$24							800	7,210
Total		3	11	19	27	621	4,985	16,725
Max.Len		4	8	10	12	22	28	34

Architectures differ in number of layers, kernel size, and number of channels.

Favoni, AI, Müller, Schuh, Phys.Rev.Lett. 128 (2022) 032003

Sketch of proof for arbitrary Wilson loops



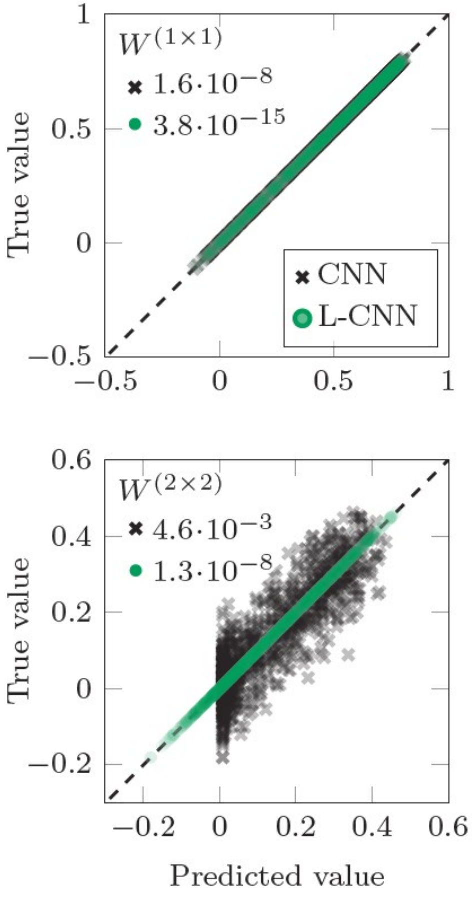
Favoni, AI, Müller, Schuh, Phys.Rev.Lett. 128 (2022) 032003

(a) An arbitrary contractible Wilson loop of *n* tiles ...
(b) … is composed (L-Bilin) of a Wilson loop of (*n*-1) tiles …
(c) … and a parallel-transported (L-Conv) plaquette (Plaq).

Non-contractible loops (like Polyakov loops) have to be added (Poly).

Numerical results

1+1D



Regression task to learn value of rectangular Wilson loops:

$$W_{x,\mu\nu}^{(m\times n)} = \mathrm{Re}\,\mathrm{Tr}\left[U_{x,\mu\nu}^{(m\times n)}\right]$$

Lattice gauge equivariant CNN (L-CNNs, green) can learn the relation, while traditional convolutional neural networks (CNNs, black) struggle to find the solution.

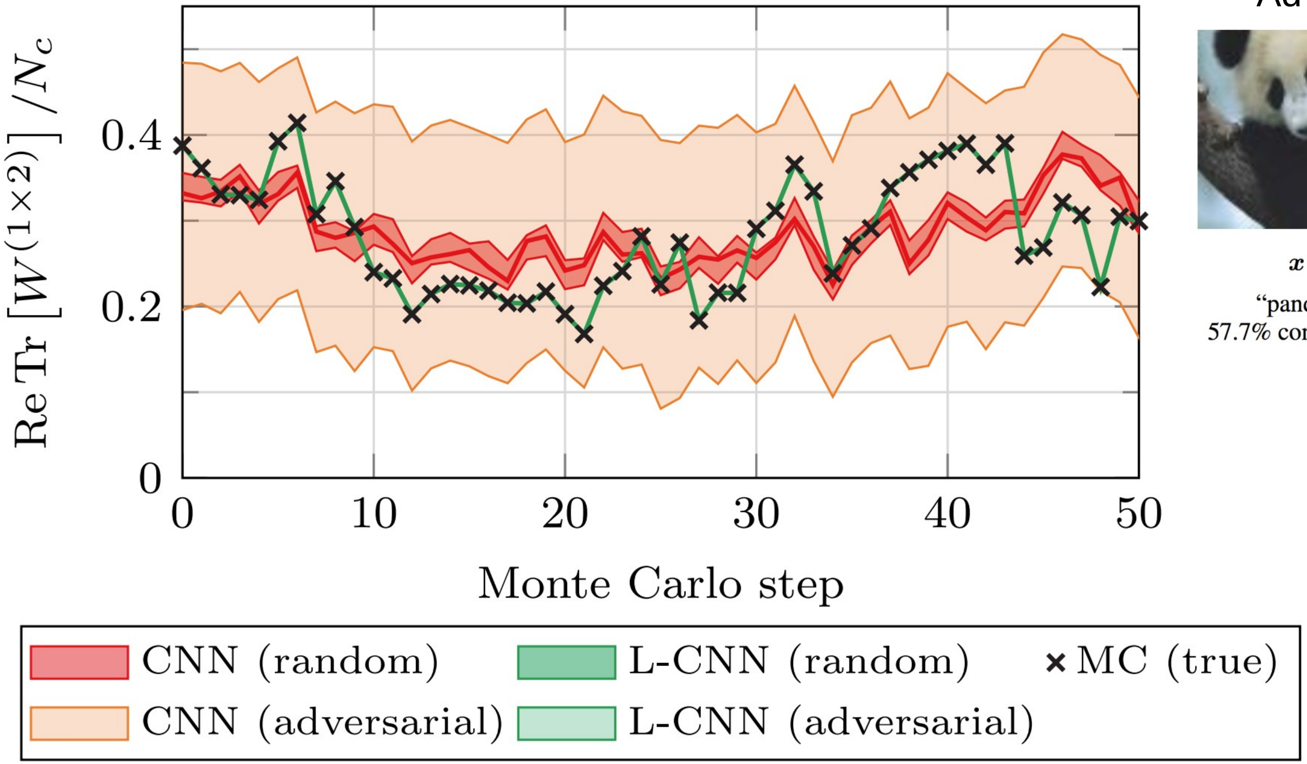
Training on 8 × 8, testing from 8 × 8 up to 64 × 64

Compared best from:
100 L-CNN models (10 – 10^4 trainable parameters, up to 4 L-Conv+L-Bilin)

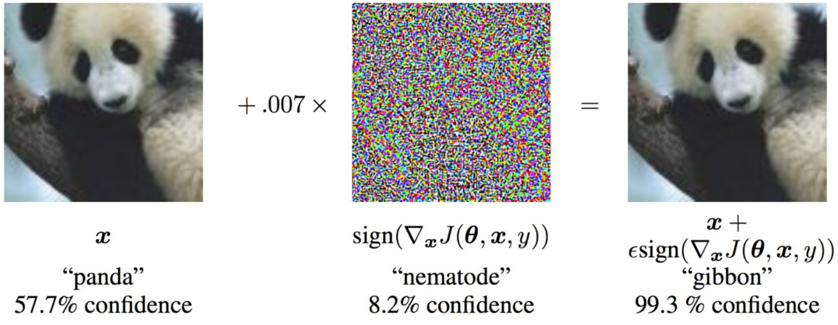
2840 CNN models (100 – 10^5 trainable parameters up to 6 layers, 512 channels, 4 activation functions)

Favoni, AI, Müller, Schuh, Phys.Rev.Lett. 128 (2022) 032003

Adversarial attacks



Adversarial attack:

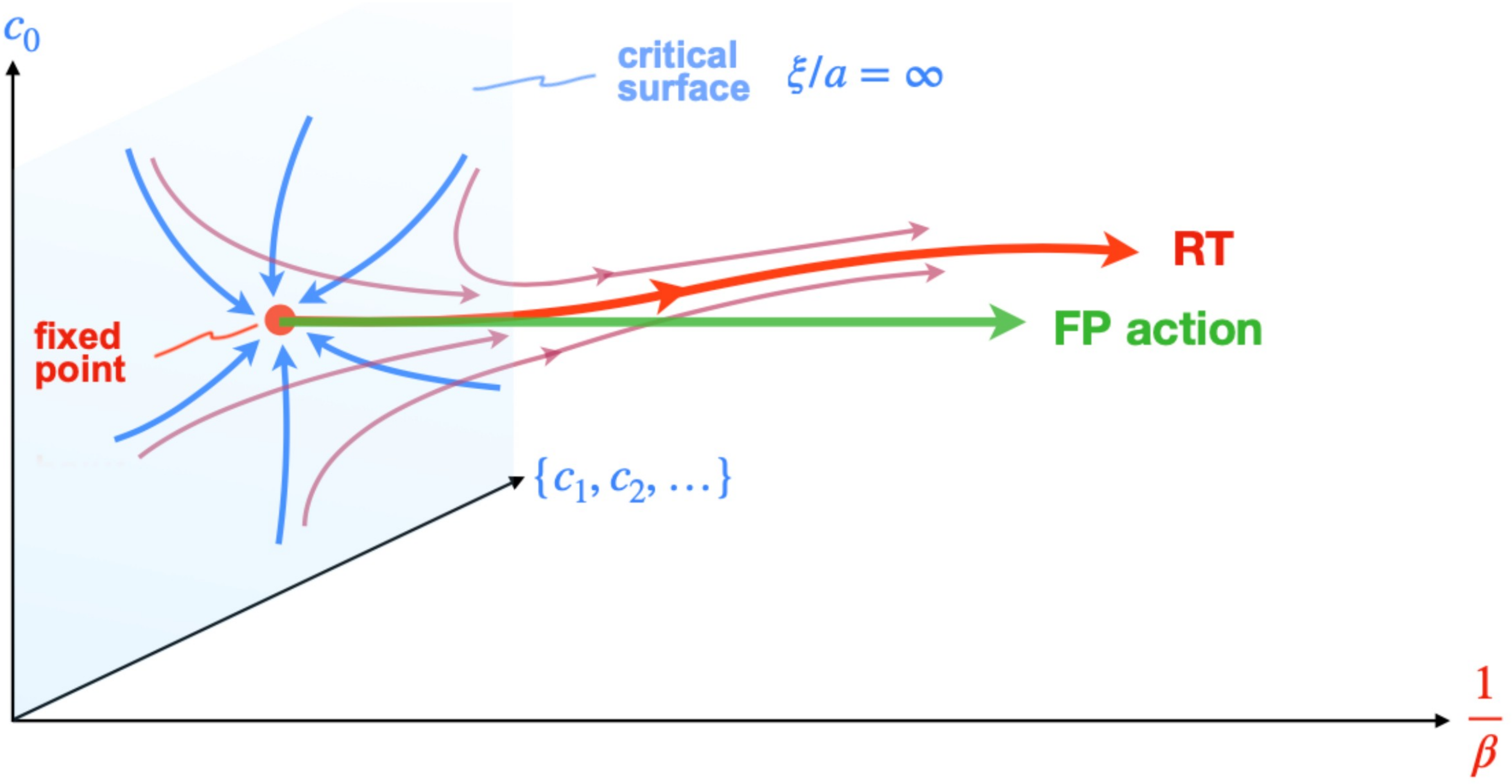


From Goodfellow, Shlens, Szegedy ICLR 2015

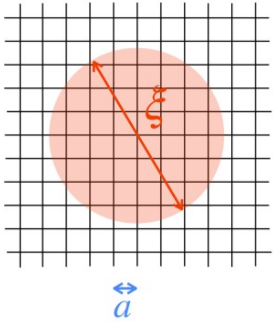
L-CNNs are insensitive to random or adversarial gauge transformations

Favoni, AI, Müller, Schuh, Phys.Rev.Lett. 128 (2022) 032003

Fixed point action



Quantum field theory on a lattice



Lattice spacing a

Partition function for SU(N_c) gauge theory

$$Z(\beta) = \int \mathscr{D}U \exp\{-\beta A[U]\}$$

Action

with gauge coupling $\beta = \frac{2N_c}{g^2}$

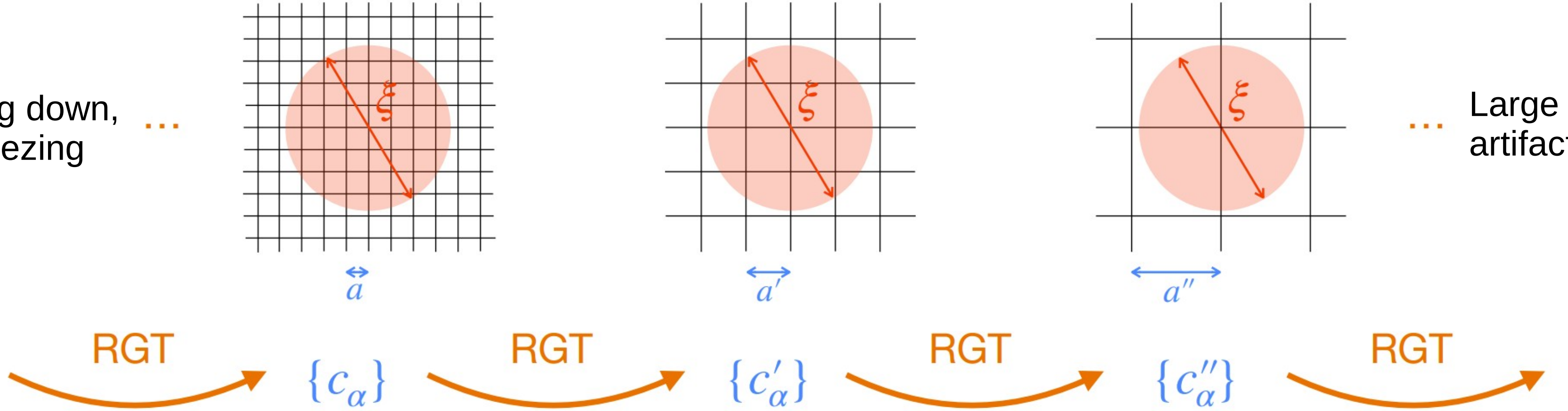
Expectation values of observables can be calculated as

$$\langle \mathcal{O}_\xi(\beta) \rangle = \frac{1}{Z} \int \mathscr{D}U \exp\{-\beta A[U]\}\, \mathcal{O}_\xi[U]$$

for a characteristic length scale ξ

Renormalization group transformation

Critical slowing down, topological freezing ...



... Large lattice artifacts

Introduce a (real space) renormalization group transformation (RGT)

$$\exp\left\{-\beta' A'[V]\right\} = \int \mathscr{D}U \exp\left\{-\beta\left(A[U] + T[U,V]\right)\right\}$$

Blocking kernel

The effective action $\beta' A'[V]$ is described

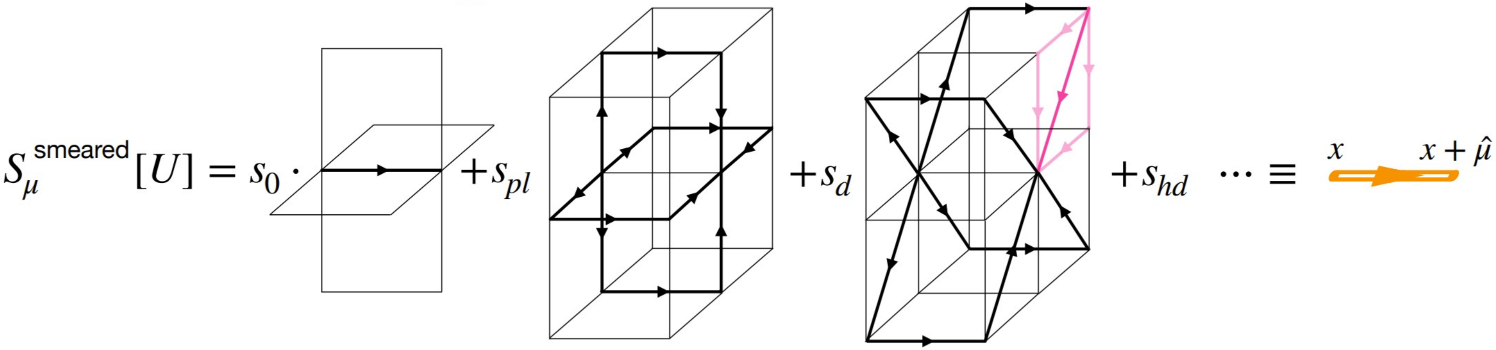
by infinitely many couplings $\{c'_\alpha\}$

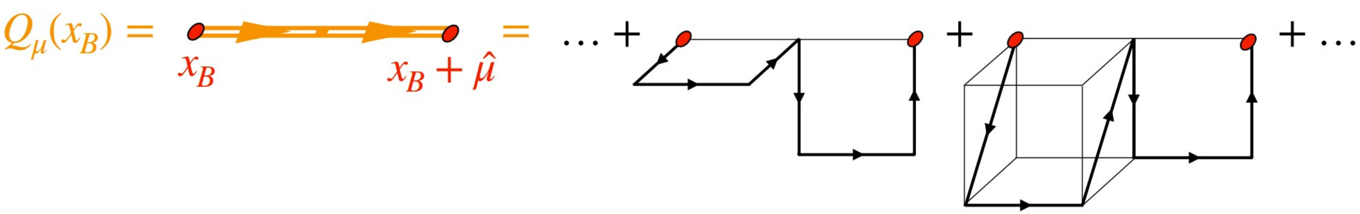
The fixed point is the saddle point in the classical limit $\beta \to \infty$, which can be found by a minimization condition.

P. Hasenfratz, F. Niedermayer, Nucl.Phys.B 414 (1994) 785

Blocking kernel

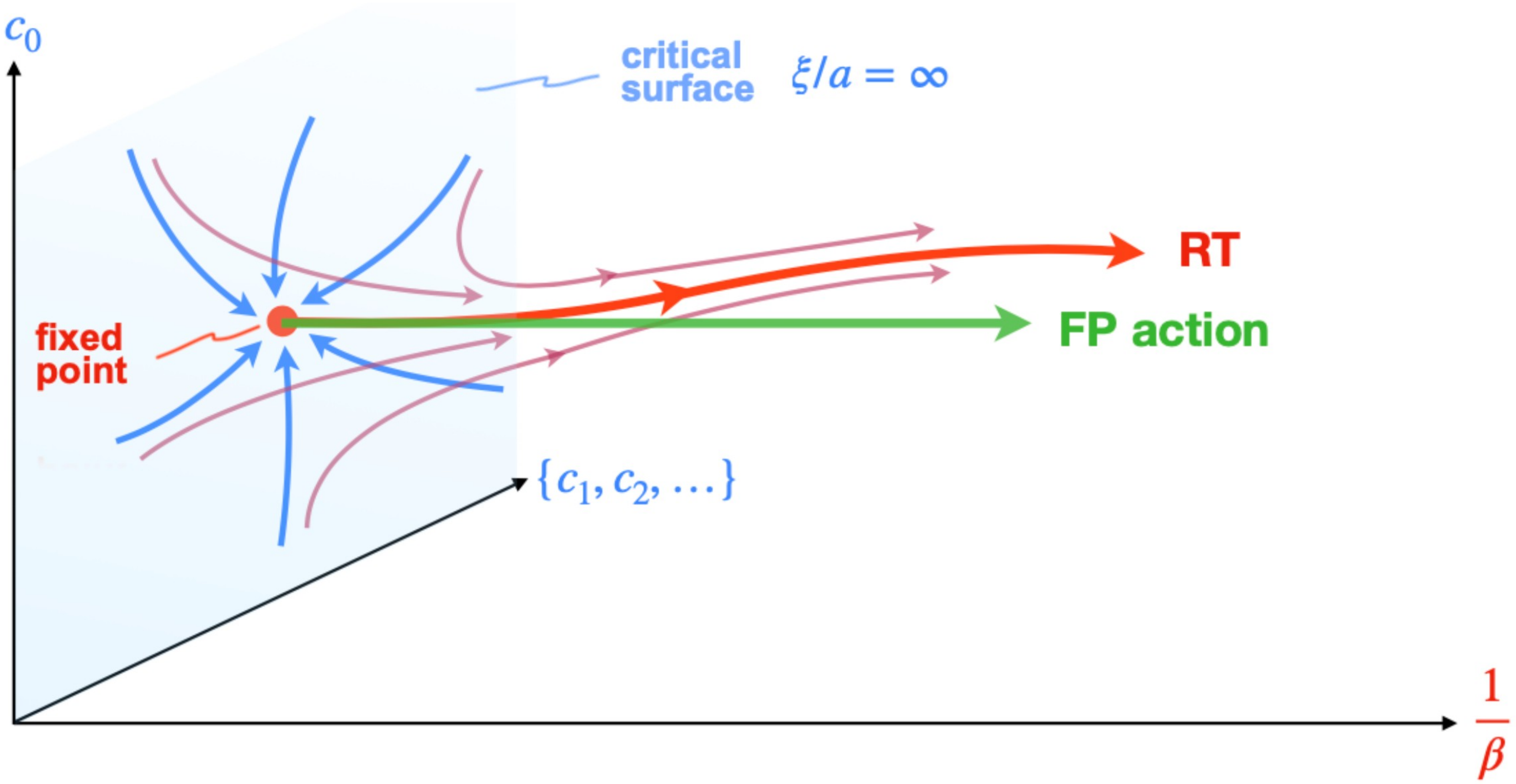
$$T[U,V] = -\frac{\kappa}{N_c}\sum_{x_B,\mu}\left\{\mathrm{ReTr}\left(V_\mu(x_B)\cdot Q_\mu^\dagger(x_B)\right) - \mathcal{N}_\mu^\beta\right\}$$





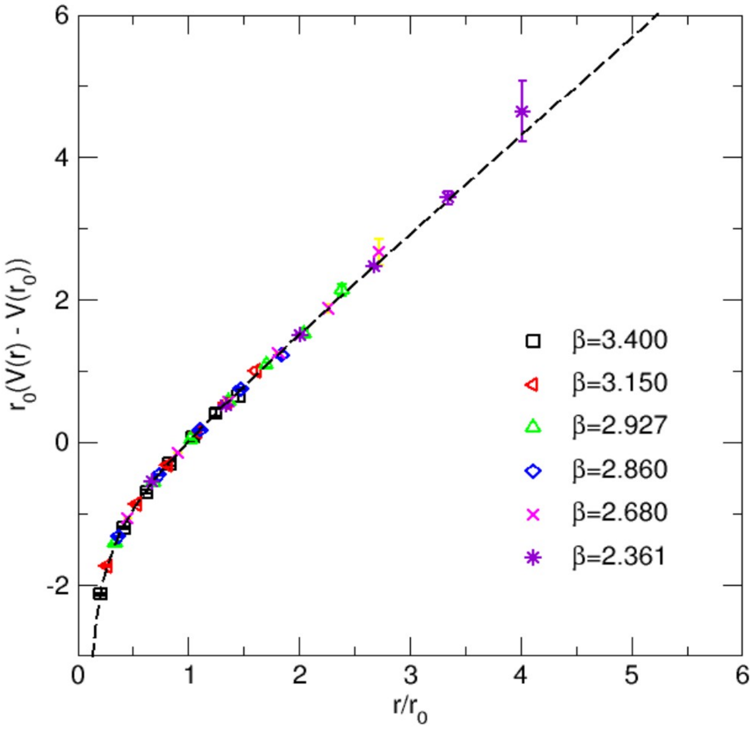
Choice of blocking kernel determines how couplings are modified across scales.

Renormalization group transformation and Fixed point action



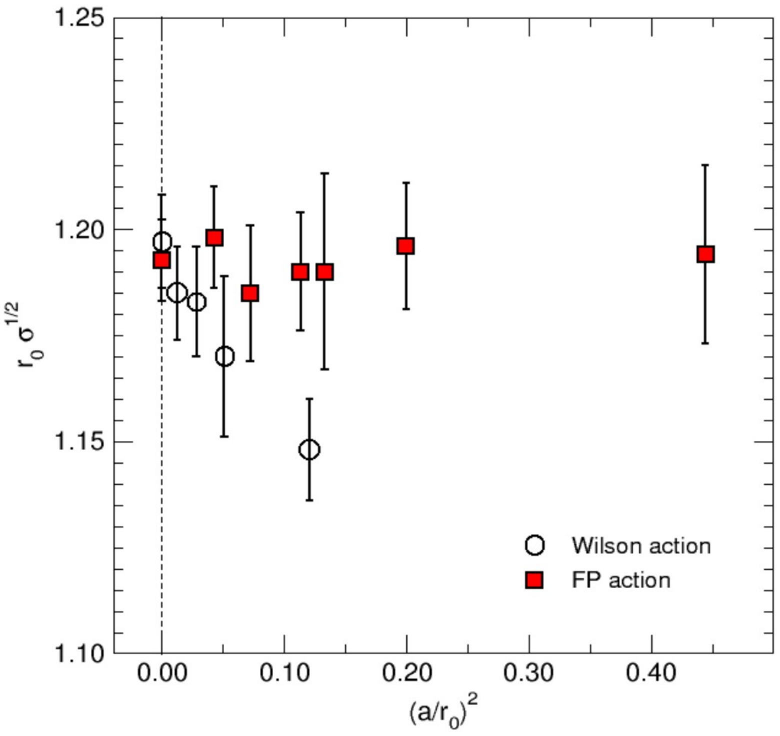
Fixed point action using older parametrizations

Static quark-antiquark potential



fit to $V(r) = V_0 - \frac{\alpha}{r} + \sigma r$

Extraction of string tension σ



Niedermayer, Rüfenacht, Wenger, Nucl.Phys.B 597 (2001) 413, hep-lat/0007007

Machine learning the fixed point action

To obtain the training data for supervised machine learning, first generate ensembles of coarse configurations *V*.

For a given coarse configuration *V*, the fixed point action values are determined by minimizing configurations *U*, *U'*, ...

$$A^{\mathrm{FP}}[V] = \min_{\{U\}} \{A^{\mathrm{FP}}[U] + T[U,V]\} = \min_{\{U',U\}} \{A^{\mathrm{FP}}[U'] + T[U',U] + T[U,V]\}$$

Use additional information for training obtained from derivatives of the fixed point action:

$$\frac{\delta A^{\mathrm{FP}}[V]}{\delta V^a_{x,\mu}} = \frac{\delta T[U,V]}{\delta V^a_{x,\mu}} = -\kappa \,\mathrm{Re}\,\mathrm{Tr}(it^a V_{x,\mu} Q^\dagger_{x,\mu}) \qquad Q^\dagger_{x,\mu} = Q^\dagger_{x,\mu}[U]$$

yields *D* [link directions] x (N^2 - 1) [colors] x L^D [lattice sites] data per configuration.

Fixed point action using L-CNNs

Parametrize **action model** in particular way:

$$\mathcal{A}^{\text{L-CNN}}[V] = \sum_x \mathcal{A}_x^{\text{pre}}[V] \sum_{n=0}^{\infty} b^{(n)} \left(N_x[V] - N_x[\mathbb{1}]\right)^n$$

Prefactor controls continuum behavior

L-CNN

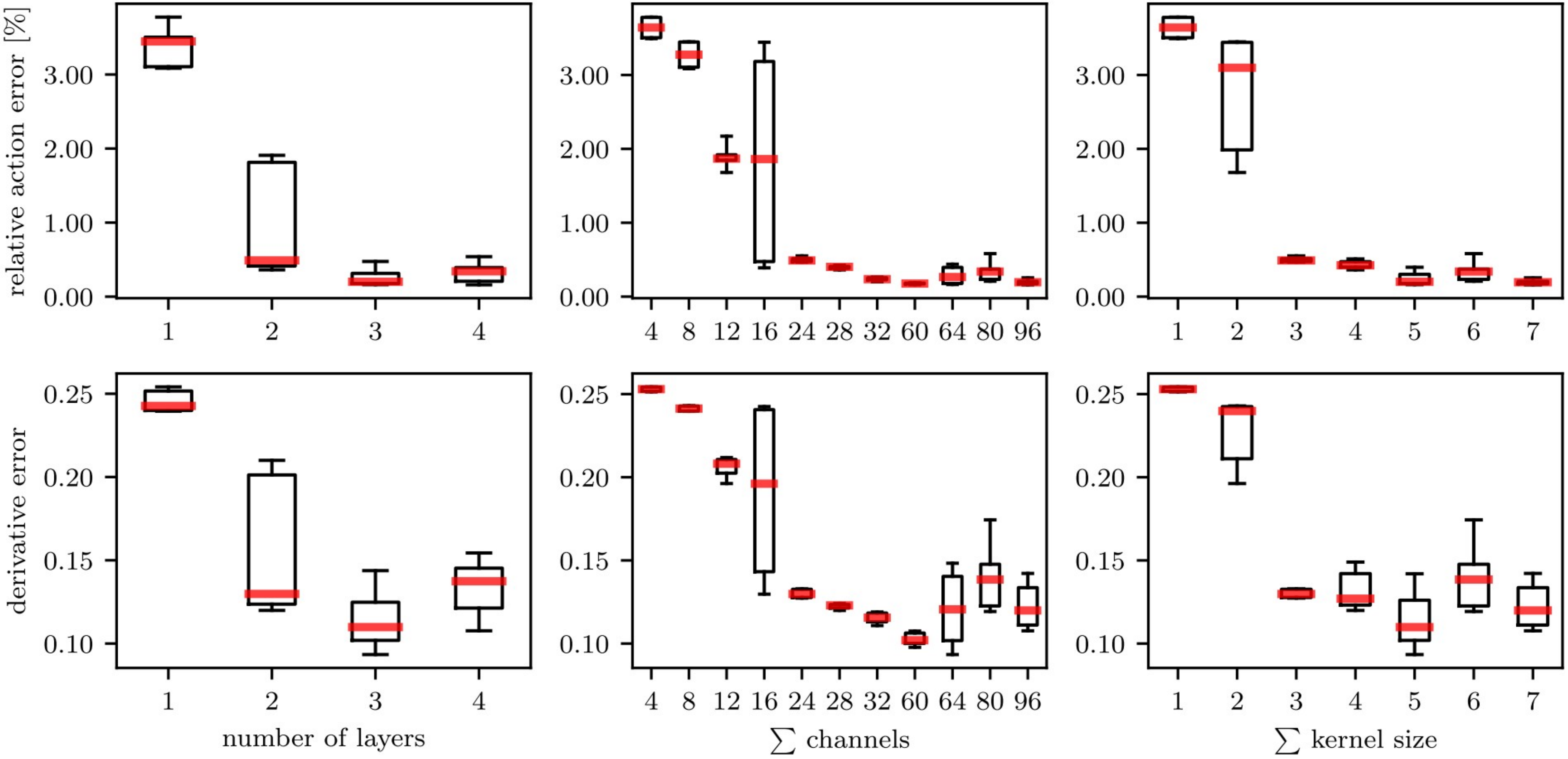
Loss function combines action values and its derivatives $\mathcal{L} = w_1 \mathcal{L}_1 + w_2 \mathcal{L}_2$

$$\mathcal{L}_1 = \frac{1}{L^4 N_{\text{cfg}}} \sum_{i=1}^{N_{\text{cfg}}} \left| \mathcal{A}^{\text{FP}}[V_i] - \mathcal{A}^{\text{L-CNN}}[V_i] \right|,$$

$$\mathcal{L}_2 = \frac{1}{32 L^4 N_{\text{cfg}}} \sum_{i=1}^{N_{\text{cfg}}} \sum_{x,\mu} \text{Tr}\left[\left(D_{x,\mu}^{\text{FP}}[V_i] - D_{x,\mu}^{\text{L-CNN}}[V_i] \right)^2 \right]$$

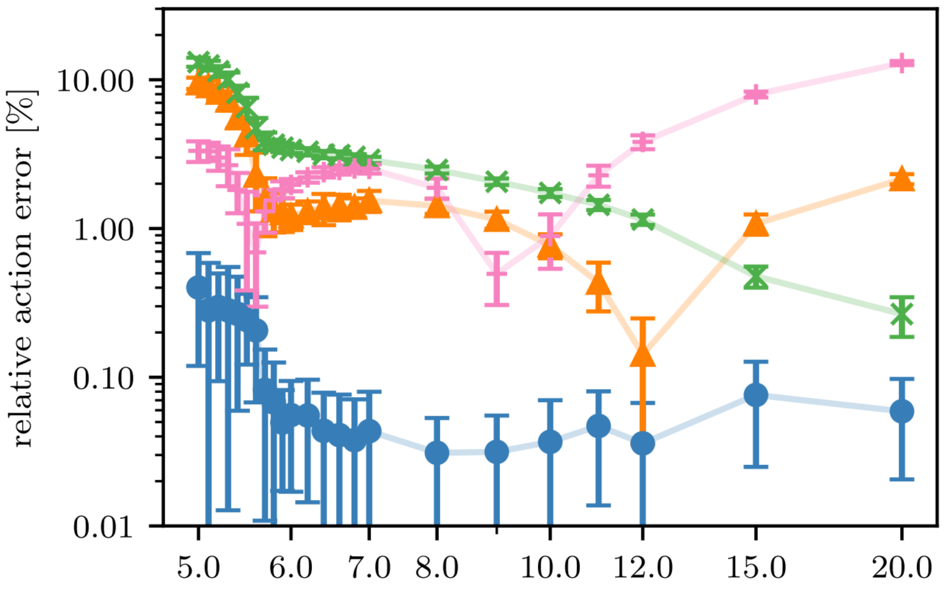
Technical remark: derivatives of L-CNNs are obtained through backpropagation

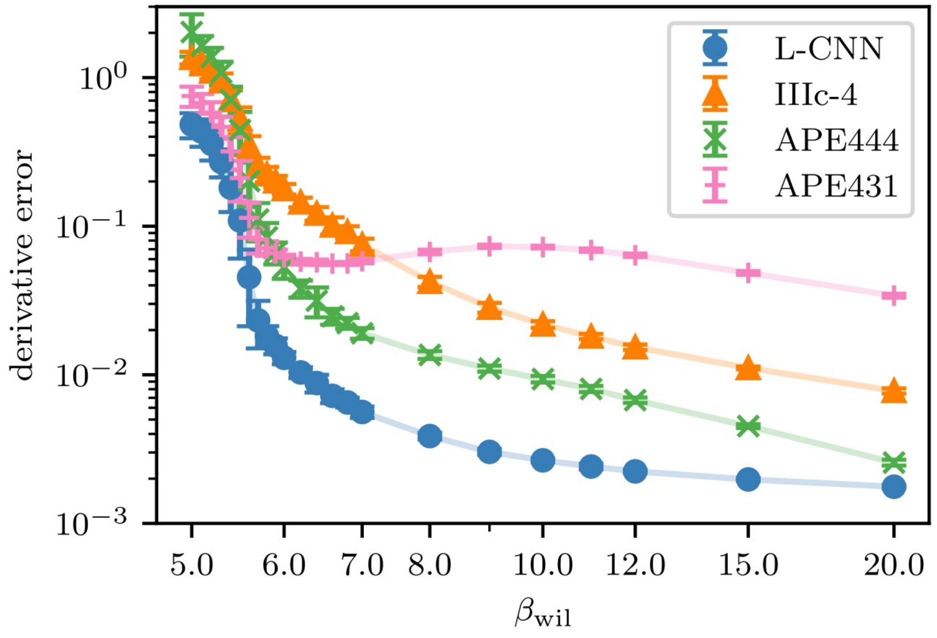
Scan through various architectures



Train 130 models of various sizes for 4^4 lattice, SU(3) gauge group, and $\beta_{wil} \in [5,10]$.

Learning the fixed point action with L-CNNs





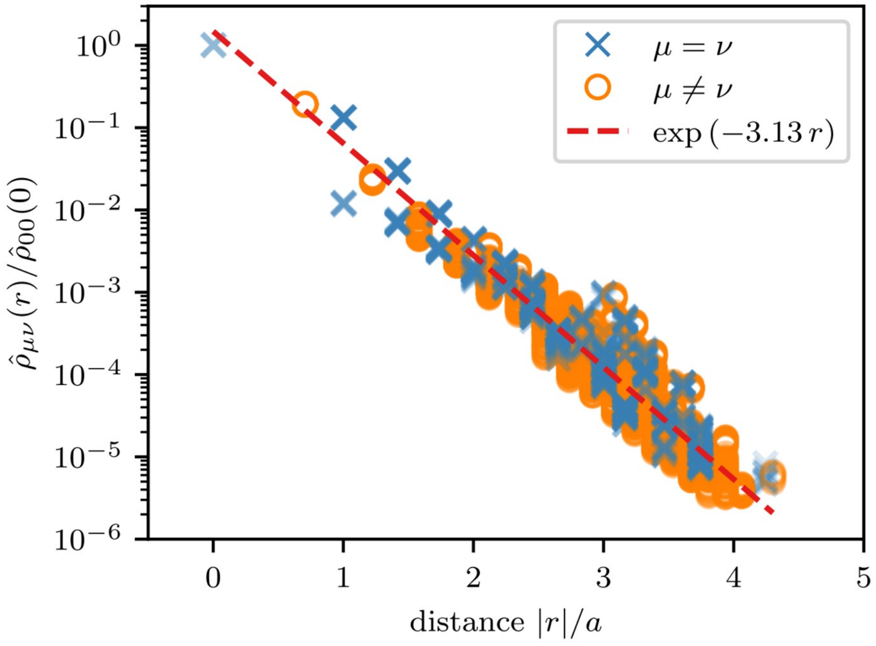
Best model: L-CNN with
3 layers with 12, 24, 24 channels
and kernel size 2, 2, 1.

L-CNN superior to older parametrizations of FP action.

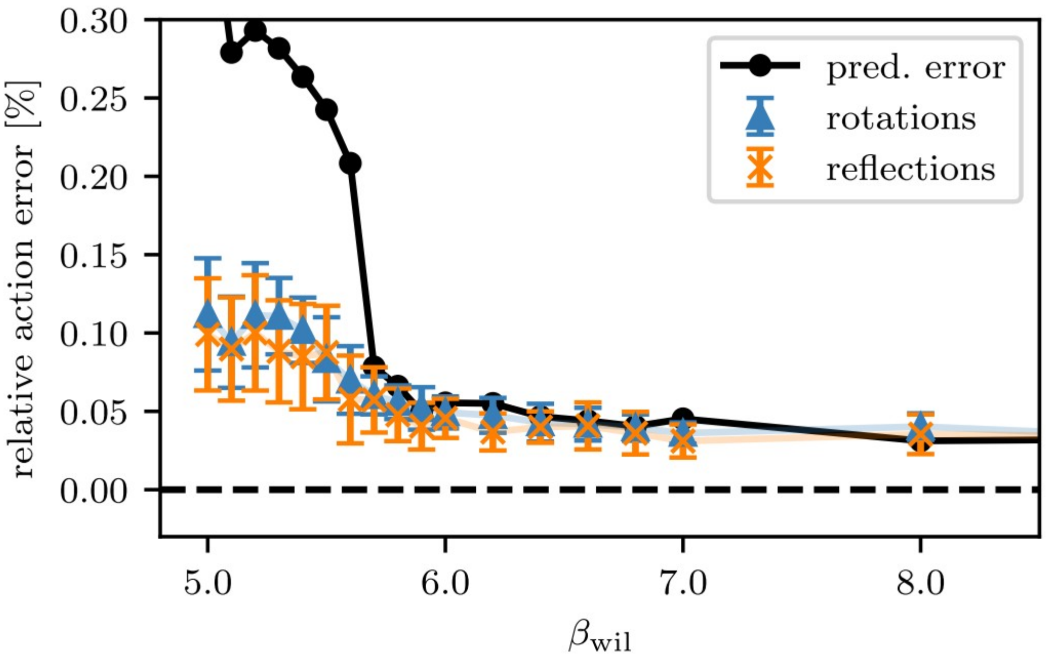
Holland, AI, Müller, Wenger
arXiv:2401.06481

Properties of the learned FP action

Locality measurement



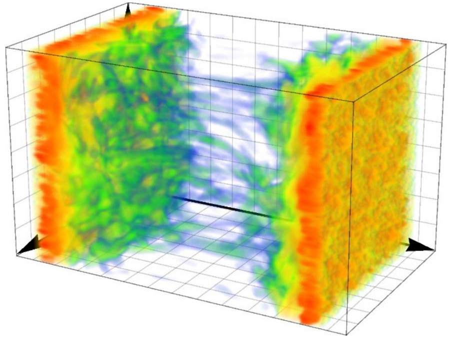
Rotation and reflection error



Holland, AI, Müller, Wenger
arXiv:2401.06481

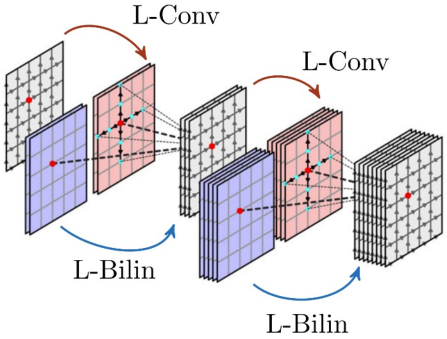
Summary

Glasma simulations



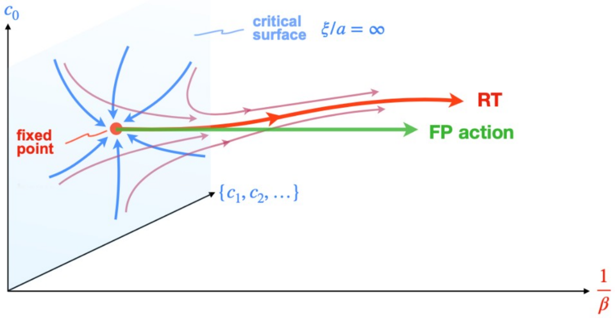
AI, Müller, Phys. Lett. B 771 (2017) 74
Gelfand, AI, Müller, Phys. Rev. D94 (2016) no.1, 014020

L-CNNs



Favoni, AI, Müller, Schuh,
Phys.Rev.Lett. 128 (2022) 032003

Learning fixed-point actions



Holland, AI, Müller, Wenger
arXiv:2401.06481

- improved gauge actions may enable simulations at coarser lattice spacing
- fixed point actions have no lattice artifacts at the classical level (if the object studied can be resolved), and may have suppressed lattice artifacts at the quantum level.
- L-CNNs achieve higher accuracy than previous hand-crafted parametrizations
- Outlook: apply learned FP action to Monte Carlo simulations.

Open source: https://gitlab.com/openpixi/lge-cnn

Backup

Continuous formulation of L-CNNs

Define a continuous version of a gauge equivariant convolution:

$$[\psi * \mathcal{W}]^a(\mathbf{x}) = \sum_b \int_{\mathbb{R}^D} \mathrm{d}\mathbf{y}^D \, \psi^{ab}(\mathbf{y}-\mathbf{x}) U_{\mathbf{x}\to\mathbf{y}} W^b(\mathbf{y}) U^\dagger_{\mathbf{x}\to\mathbf{y}}$$

with kernel components $\psi^{ab} : \mathbb{R}^D \to \mathbb{R}$

and parallel transporter $U_{\mathbf{x}\to\mathbf{y}} = \mathcal{P} \exp \left\{ i \int_0^1 \mathrm{d}s \frac{\mathrm{d}x^\nu(s)}{\mathrm{d}s} A_\nu(x(s)) \right\}$

that map $\mathcal{W} = (W^1, \dots, W^m)$ objects to new objects

in a gauge equivariant manner:

$$[\psi * T_\Omega \mathcal{W}]^a(\mathbf{x}) = T_\Omega [\psi * \mathcal{W}]^a(\mathbf{x})$$

Similarly define continuous bilinear layer, trace layer, …

Discretize this to obtain previous formulation.

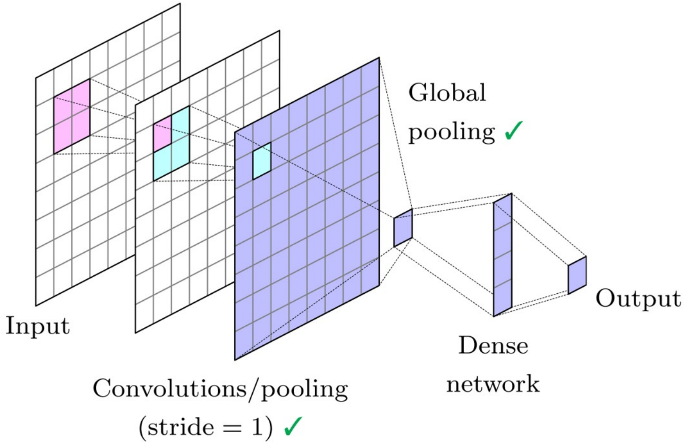
Compatible with G-CNNs.

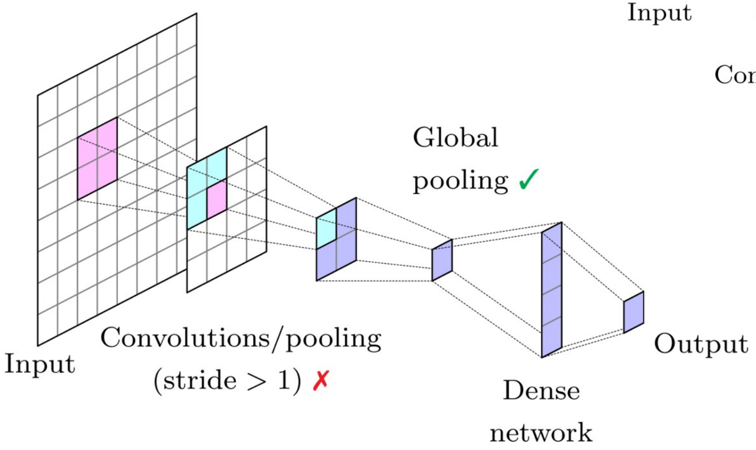
Generalizable to vectors and tensors.

Aronsson, Müller, Schuh, arxiv:2303.11448

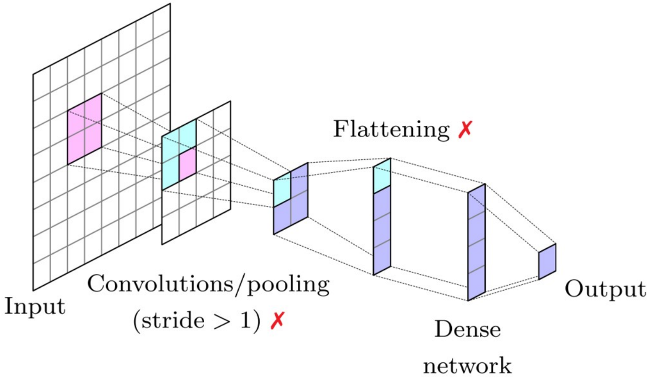
Translational symmetry

Equivariant architecture (EQ)





Strided architecture (ST)



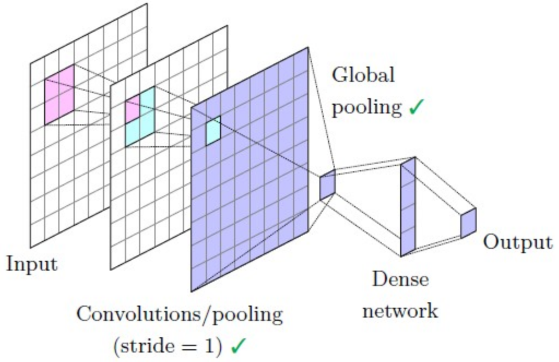
Flattening architecture (FL)

Bulusu, Favoni, AI, Müller, Schuh, Phys. Rev. D 104 (2021) 074504

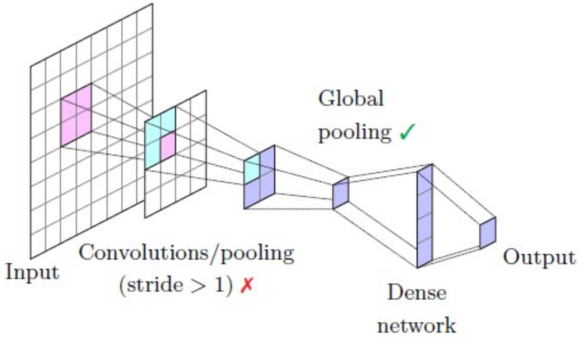
Comparison of architecture types

For fair comparison, best architectures for each type have been obtained by an Optuna optimization (scanning through various kernel sizes, number of layers, number of channels, …)

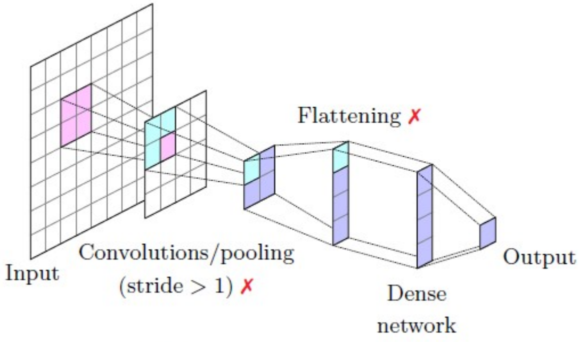
Best architectures are retrained 10 times and evaluated on the validation set.



(a)Equivariant architecture (EQ)

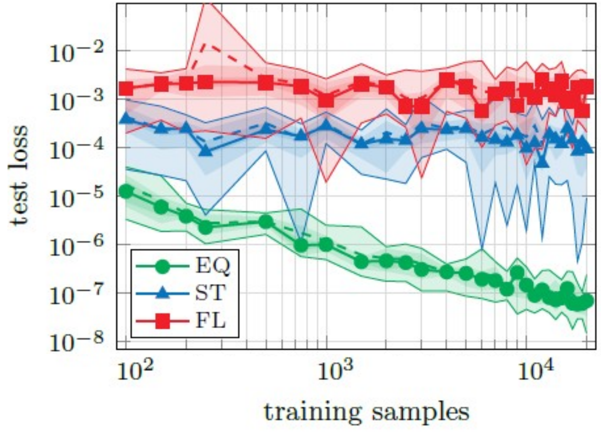


(b)Strided architecture (ST)



(c)Flattening architecture (FL)

Test regression tasks on observables of a scalar field model in 2 dimensions:



Bulusu, Favoni, AI, Müller, Schuh, Phys. Rev. D 104 (2021) 074504