BERGISCHE UNIVERSITÄT WUPPERTAL

Localized machine learned flow maps to accelerate Markov Chain Monte Carlo simulations

29. May 2024

Trento - Workshop

Machine learning and the Renormalization Group

Jacob Finkenrath

FOR 5269

Table of Content

Motivation/Introduction

- Standart Model of Particle Physics
- Monte Carlo simulation of lattice gauge theories

Gauge equivariant/normalizing flows

- Proposal via flows
- Domain Decomposition

J. F., arXiv:2201.02216

Fine graining flows in 2D

- Maps and training
- Tunneling rate

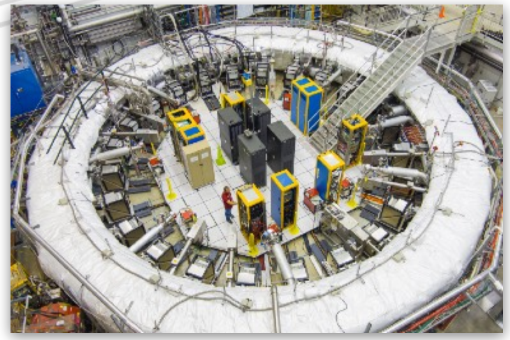
J. F., arXiv:2402.12176

Global corrections with the fermion determinant

- Domain decomposition of fermions
- Towards high acceptances

FOR 5269

Motivation: Lattice QCD at the Precision Frontier

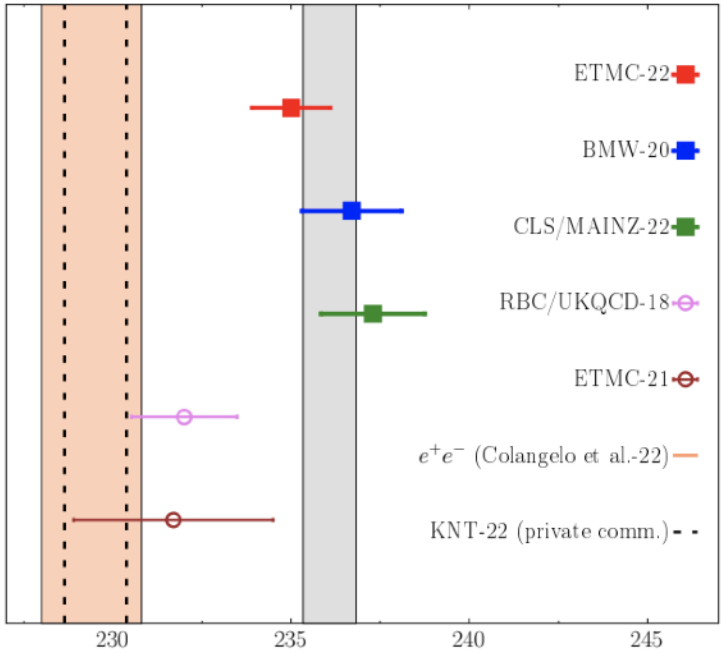


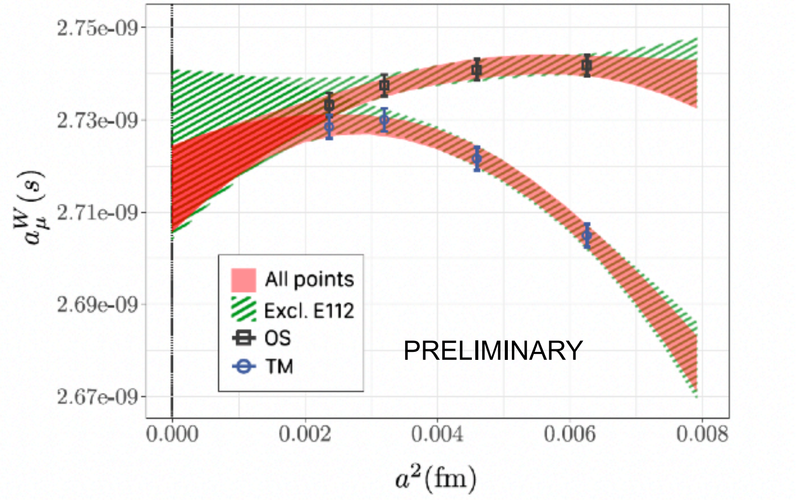
Lattice Quantum Chromodynamics
At the precision frontier

Muon and Flavor Physics
are indicating New Physics;
ab initio LQCD calculations are needed

Search for new physics in the precision frontier by
- ❖ high precision measurements
- ❖ theoretical prediction

deviations are signs for new physics





Anomalous magnetic moment of muon:
Muon g-2 Experiment at FermiLab increased precision
- ❖ > 5σ deviation between experiment and data-driven approach
- ❖ 4σ deviation between lattice and data-driven approach

To resolve this puzzle:

Precision Measurement of Lattice QCD are needed

❖ finer lattice spacings needed to match future experiments precision

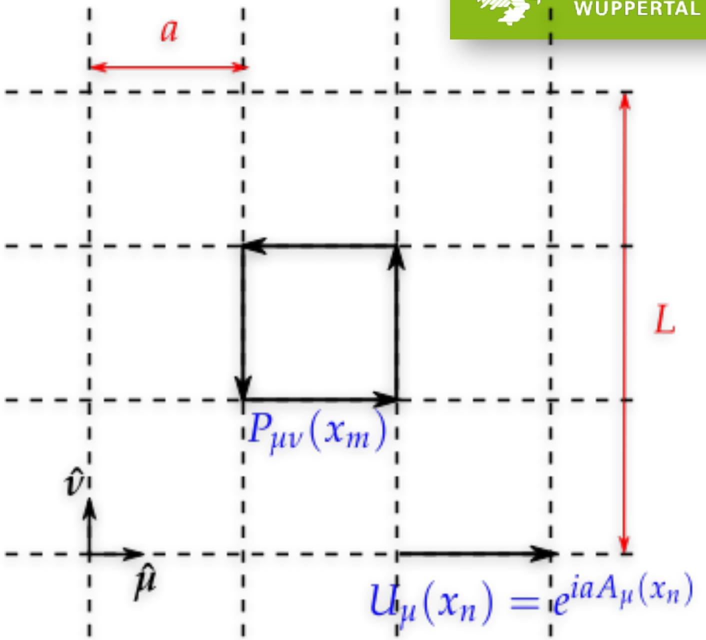
BERGISCHE UNIVERSITÄT WUPPERTAL

Simulation at the Precision Frontier: Markov chain Monte Carlo

$$\langle \mathcal{O} \rangle = \int D[U, \phi]\, \mathcal{O}(U) \cdot \rho(U)$$

where

$$\rho(U) = Z^{-1} \left(\prod_j^{N_f} \det D_j(U) \right) e^{-\beta_g(U)}$$



Lattice QCD:
❖ 4 dimensional lattice: V=L x L x L x T
❖ State of the art V = 2* (96^4) points

❖ physical degrees of freedoms:
SU(3) matrices, 8 real numbers per matrix

❖ SU(3) matrices are acting as parallel transporter between points:
-> 4*V links

High dimensional integral
❖ possible to solve via Markov chain Monte Carlo methods

here, we will mainly use:

We will discuss mainly the 2D-Schwinger model with U(1) links

Generic models are only applied to *pure gauge weight*

$$\rho_{PG} \propto \exp\{-\beta S_g(U)\}$$

And fermions are treated via correction steps

$$\rho_f \propto \prod_j^{N_f} \det D_j(U)$$

FOR 5269

BERGISCHE UNIVERSITÄT WUPPERTAL

here: 2D Schwinger Model

Markov chain Monte Carlo algorithm

Standard large scale MCMC method:

- Hybrid Monte Carlo (HMC) algorithm
  - based on molecular dynamics

$$\dot{P} = -\frac{\partial H}{\partial U} \quad \text{and} \quad \dot{U} = \frac{\partial H}{\partial P}$$

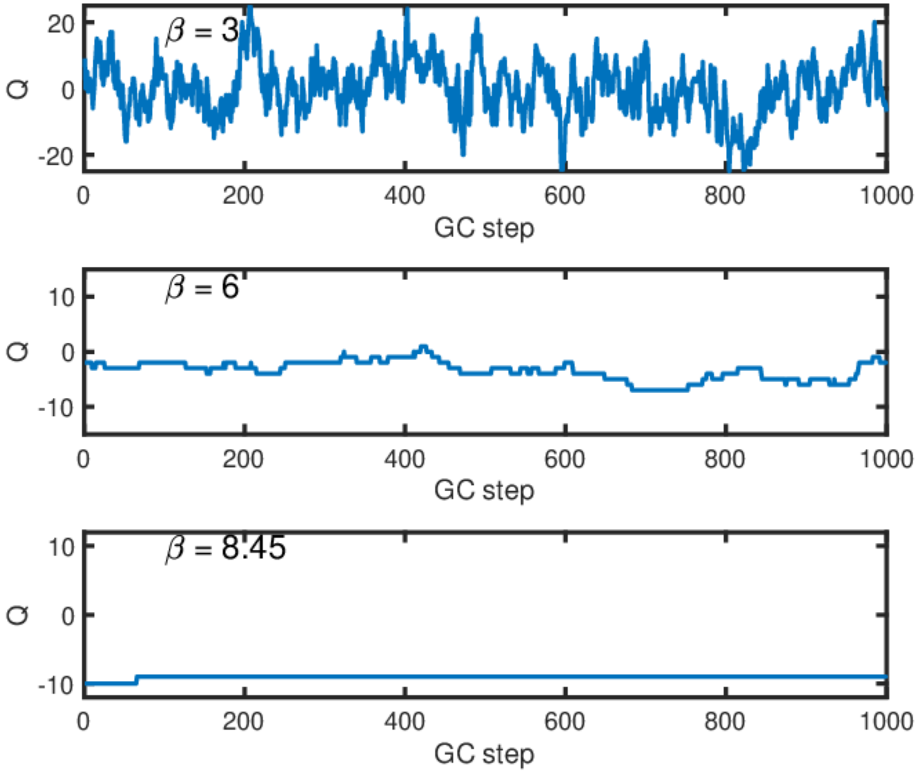
❖can be integrated using numerical integrators
❖Sampling configuration in field space

❖Ensemble average over these configurations:

$$\langle \mathcal{O} \rangle = \frac{1}{N} \sum_i \mathcal{O}(U_i) + \sqrt{\frac{2\tau_{int}\sigma^2}{N}}$$

❖ for very fine lattice spacings a<0.05 fm sampling of independent configuration becomes hard $\tau_{int} \gg 1$
❖ the HMC algorithm freezes out a topological sector

S. Schaefer et al., Null. Phys. B 845 (2011) 93-119



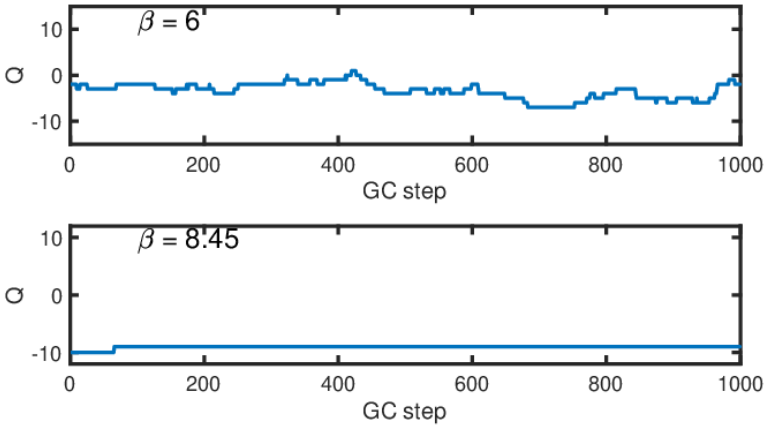
severe critical slowing down

- Efficient algorithm in QCD missing (openBC would be a possibility)

BERGISCHE UNIVERSITÄT WUPPERTAL

General structure of a MCMC algorithm:

1. Propose U' according to $T_0(U \to U')$
2. Correct with $P_{acc}(U \to U') = \min\left[1, \frac{\tilde{\rho}(U)\rho(U')}{\rho(U)\tilde{\rho}(U')}\right]$



MCMC samples sufficient if:

❖ Proposal can efficiently propose independent configurations
❖ Correction steps has a good acceptance rate

Sampling via HMC simulations are suffering from point 1.)
❖ integrating Molecular Dynamics induces only small changes in the fields

❖ **How to improve that ?**

FOR 5269

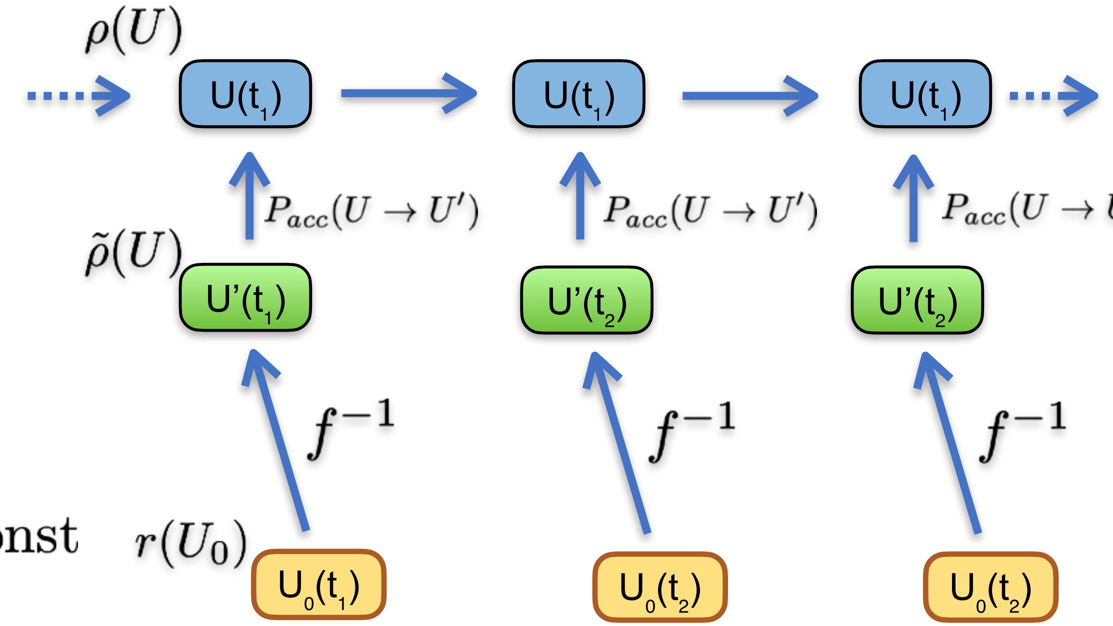
BERGISCHE UNIVERSITÄT WUPPERTAL

Idea: Sampling from an independent distribution

Idea: starting from a unitary distribution and map into target space

★**Trivialising maps**: Luescher, Commun.Math.Phys. 293 (2010)

- start from a trivial distribution $r(U_0)$
- construct field transformation towards target distribution $f^{-1}(U_0) \to U$



Flow distribution is given then by the Jacobian of the transformation

$$\tilde{\rho}(U) = r(f(U)) \cdot \left| \det \frac{\partial f(U)}{\partial U} \right|$$

Trivialising Map: $$\ln(\tilde{\rho}(f^{-1}(U_0))/\rho(f^{-1}(U_0))) = \text{const}$$

Maps can be parametrised and learned:

- ✦ Equivariant flows using neural networks
- ✦ Continuous flows Bacchio et al., arXiv:2212.08469

Here, we will use Albergo et al., Phys.Rev.D 100 (2019) 3, 034515; Kanwar et al., Phys.Rev.Lett.125 (2020) 12, 121601; Boyda et al., Phys.Rev.D 103 (2021) 7, 074504; Albergo et al., arXiv:2101.08176

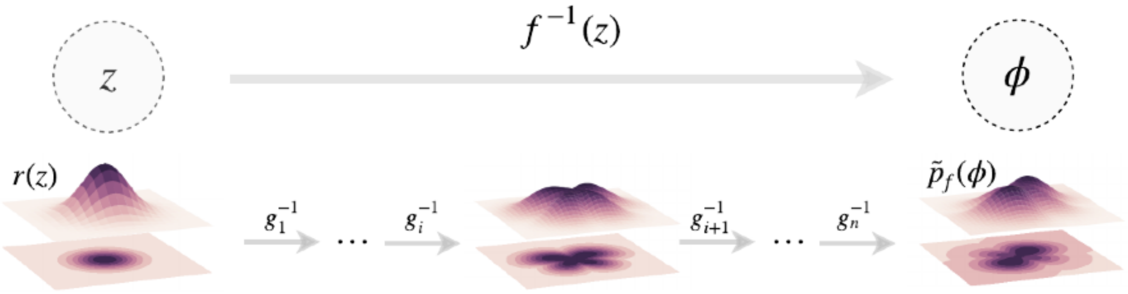
FOR 5269

Generative model

Generative model in ϕ^4 - model (U(1)) with normalising (gauge invariant) flow

Albergo et al., Phys.Rev.D 100 (2019) 3, 034515

$$\tilde{\rho}(U) = r(f(\phi)) \prod_j \det J(g_j^{-1}(\phi^{(i)}, s_i, t_i))$$



- introduce coupling layers with

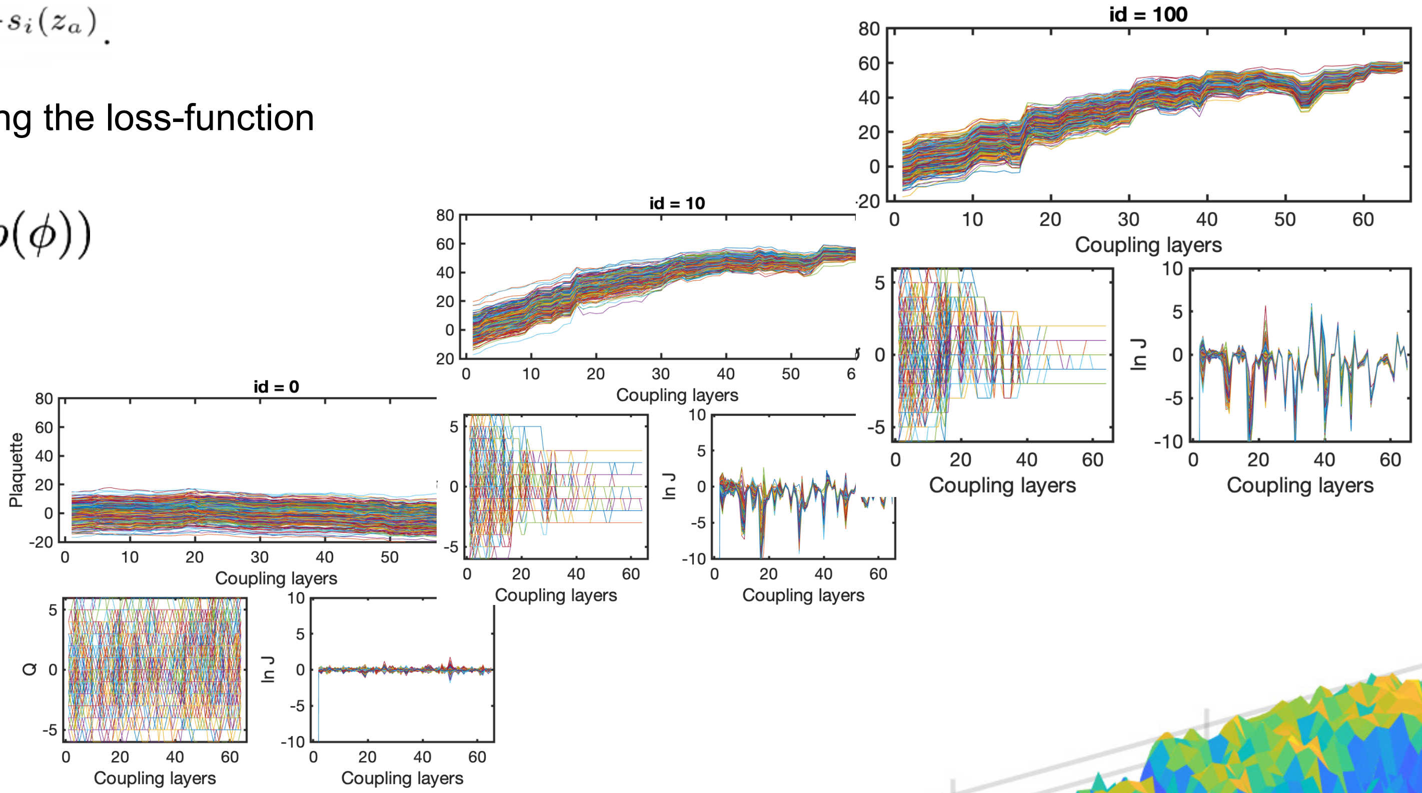
$$g_i^{-1}(z) := \begin{cases} \phi_a = z_a \\ \phi_b = (z_b - t_i(z_a)) \odot e^{-s_i(z_a)}. \end{cases}$$

- train the coupling layers t_i and s_i by minimizing the loss-function

$$L(\tilde{\rho}) = \int \prod \phi_j \, \tilde{\rho}(\phi) \ln(\tilde{\rho}(\phi)/\rho(\phi))$$

Construction of the layer such that

- Forwards and backward map easily to compute
- Alternating freezing and unfreezing variable to get block diagonal Jacobians



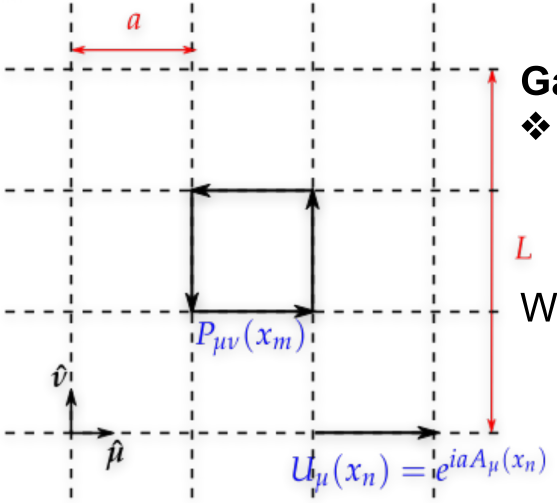
Generative models in Gauge theories

Application to gauge theories:

Kanwar et al., Phys.Rev.Lett.125 (2020) 12, 121601

Boyda et al., Phys.Rev.D 103 (2021) 7, 074504

Albergo et al., arXiv:2101.08176



Gauge invariant maps

❖ Lattice actions are gauge invariant under

$$U_\mu(x) \to g(x)^\dagger U_\mu(x) g(x+\hat{\mu})$$

Wilson pure gauge action is given by sum over plaquettes:

$$S_g(U) = 1 - \frac{1}{2N} \mathrm{ReTr} \sum_{x,\mu>\nu} P_{\mu,\nu}(x)$$

With $P_{\mu,\nu}(x) = U_\mu(x) U_\nu(x+\hat{\mu}) U_\mu^\dagger(x+\hat{\nu}) U_\nu^\dagger(x)$
Which is gauge invariant

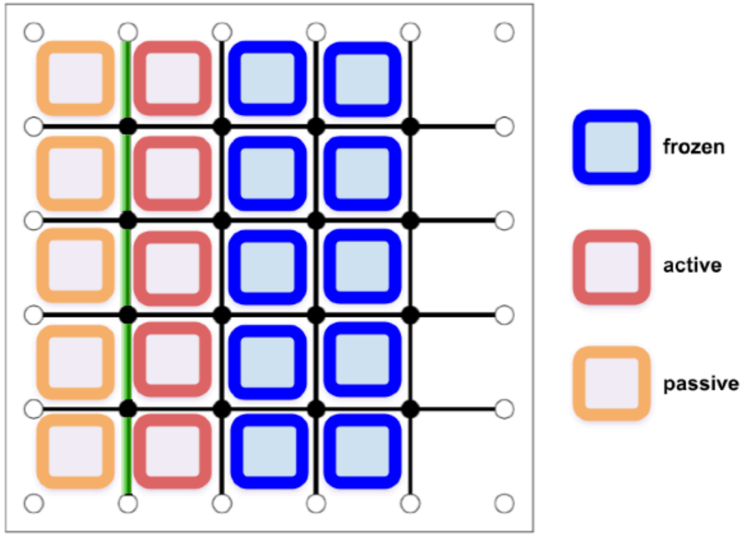
Idea: update gauge invariant object, plaquettes
And propagate update to the links via

$$U'(x) = P'_{\mu,\nu}(x) U_\nu(x) U_\mu(x+\hat{\nu}) U_\nu^\dagger(x+\hat{\mu})$$

- however this change also neighbour plaquettes

Maps are changing:

❖ *Active* : are updated
❖ *Passive* : are not touched
❖ *Frozen* : can be use to feed the networks



❖ Note: Links and plaquette needs to stay in the group
❖ Coupling layers/parameterization needs to account for that
  ❖ See for details

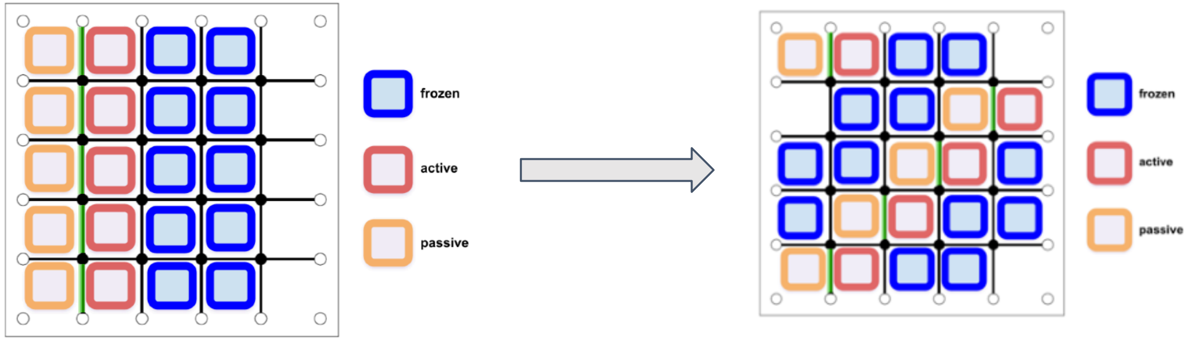
Boyda et al., Phys.Rev.D 103 (2021) 7, 074504

R. Abbott et al., arXiv:2401.10874

FOR 5269

Some insides into gauge invariant flows

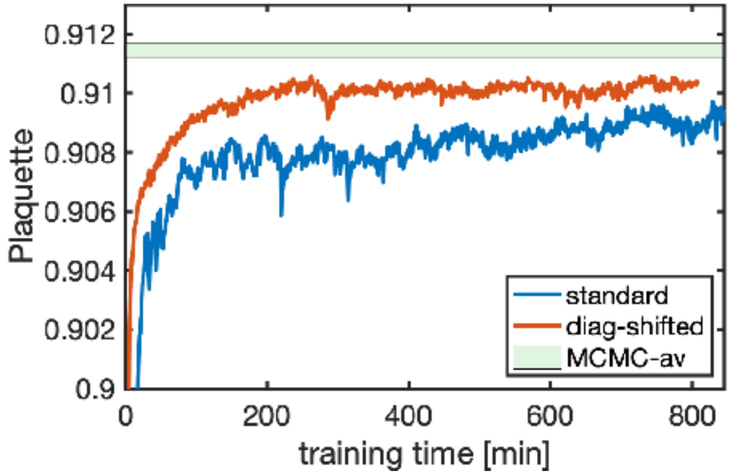
How to design coupling layers:



- Mapping can be optimized by including symmetries
  - Increase overlap with closest frozen plaquettes

Structure of networks

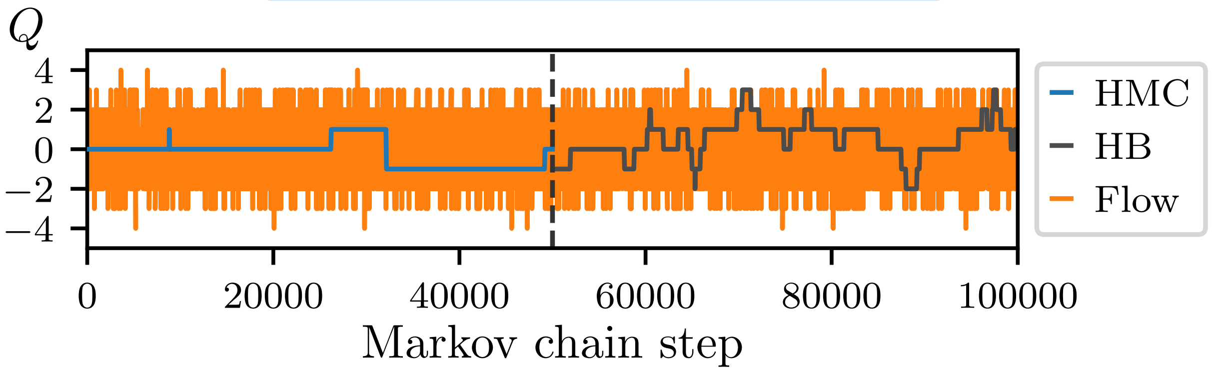
- convolutional kernels with size 3
  - note that only frozen plaquettes are used as input values
- with hidden layers (here default 2 with 8 nodes)
- 8 coupling layers corresponds to a full update



Updating masks improve convergence rate and acceptance from 30% to 50%

Decoupling of topological sectors:

Kanwar et al., Phys.Rev.Lett.125 (2020) 12, 121601



- Gauge proposal can indeed solve topological freezing

FOR 5269

BERGISCHE UNIVERSITÄT WUPPERTAL

BERGISCHE UNIVERSITÄT WUPPERTAL

Lets take a look to the second step:

1. Propose U' according to $T_0(U \to U')$
2. Correct with $P_{acc}(U \to U') = \min\left[1, \frac{\tilde{\rho}(U)\rho(U')}{\rho(U)\tilde{\rho}(U')}\right]$

We have a proposal which generates independent configurations with weight:

$$\tilde{\rho}(U) = r(f(U)) \cdot \left|\det \frac{\partial f(U)}{\partial U}\right|$$

Need correction steps to correct towards the right weight: $\rho_{PG} \propto \exp\{-\beta S_g(U)\}$

Acceptance rate:

In case ratio of distributions $(\tilde{\rho}(U)\rho(U'))/(\rho(U)\tilde{\rho}(U'))$ is log-normal distributed.

- for the acceptance rate follows

Creutz, Phys. Rev. D38 (1988) 1228–1238

$$P_{acc} = \mathrm{erfc}\{\sqrt{\sigma^2(\Delta S)/8}\}$$

with $\Delta S = \ln\{\rho(U')\} - \ln\{\rho(U)\} + \ln\{\tilde{\rho}(U)\} - \ln\{\tilde{\rho}(U')\}$

Scaling of normalizing flows

BERGISCHE UNIVERSITÄT WUPPERTAL

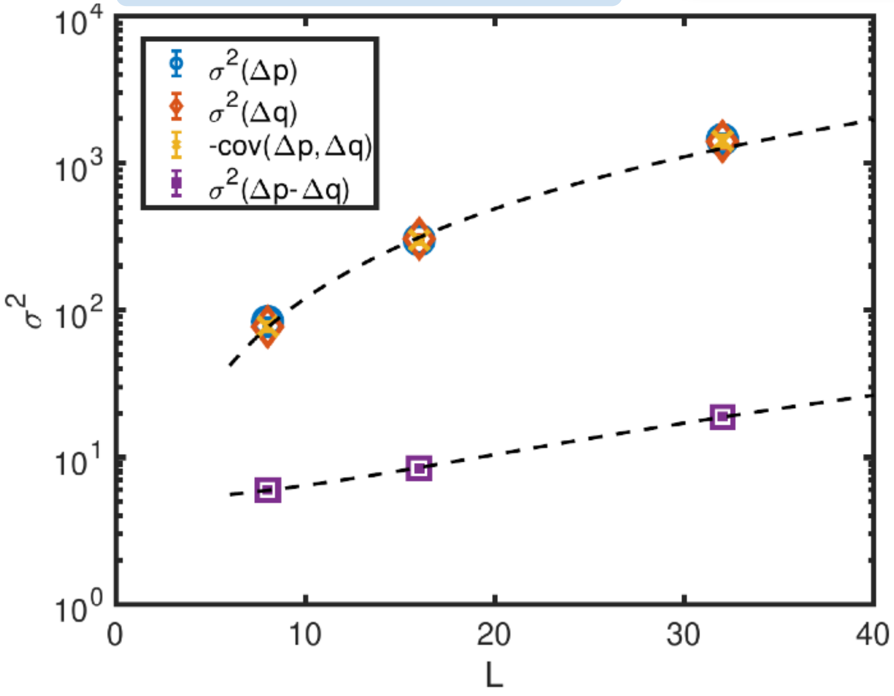
Fine tuning problem:
How the variance scales $\sigma^2(\Delta S)$

Covariances of distributions scales like variances

$$\mathrm{var}(\Delta\rho) + \mathrm{var}(\Delta\tilde{\rho}) \approx -2 \cdot \mathrm{cov}(\Delta\rho, \Delta\tilde{\rho})$$

But $\sigma^2 = \mathrm{var}(\Delta\rho) + \mathrm{var}(\Delta\tilde{\rho}) - 2 \cdot \mathrm{cov}(\Delta\rho, \Delta\tilde{\rho})$
still grows with the volume

Del Debbio, Phys. Rev. D 104, 094507

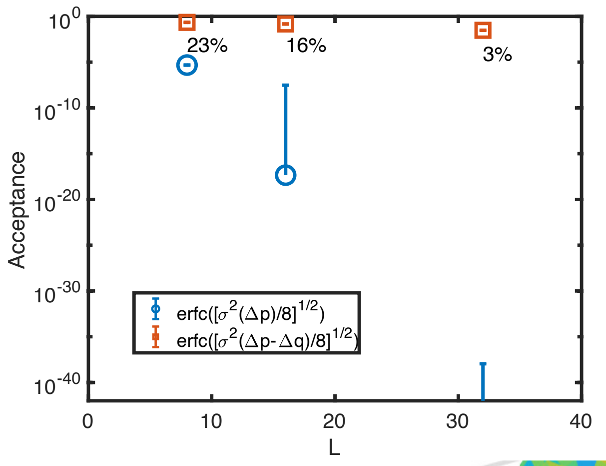


Volume fluctuations

❖ Localized models:

$$\sigma^2(S) = \langle S^2 \rangle - \langle S \rangle^2 \quad = V(a_0 + a_1 e^{-d} + a_2 e^{-\sqrt{2}d} + \ldots)$$

- ❖ Variance scales with the volume, acceptance rate is rapidly 0
- ❖ Requires modifications for larger volumes



FOR 5269

BERGISCHE UNIVERSITÄT WUPPERTAL

Idea: Decomposition of lattice into domain

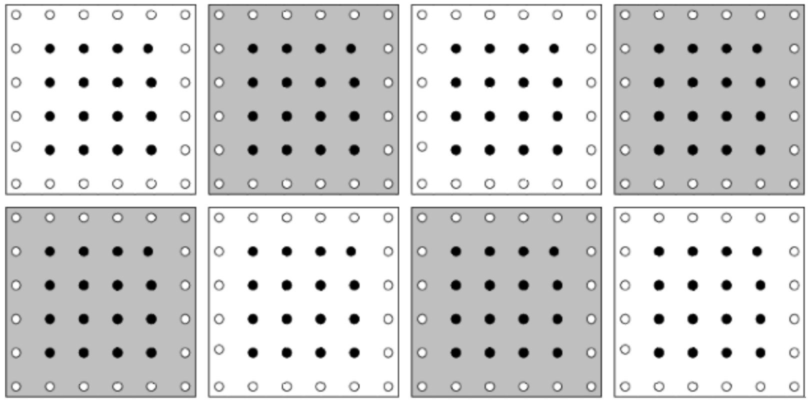
Separate action into:

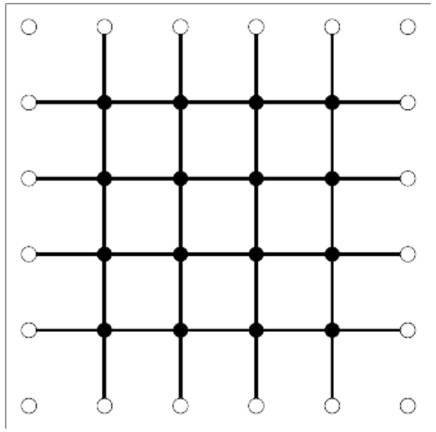
$$S_{global} = \sum_{blk} S_{local} + I(S_{global}, S_{local})$$

Decomposition straightforward for ultra local lattice actions

- ϕ^4 - model
- Pure gauge theories

This becomes harder if fermions are included





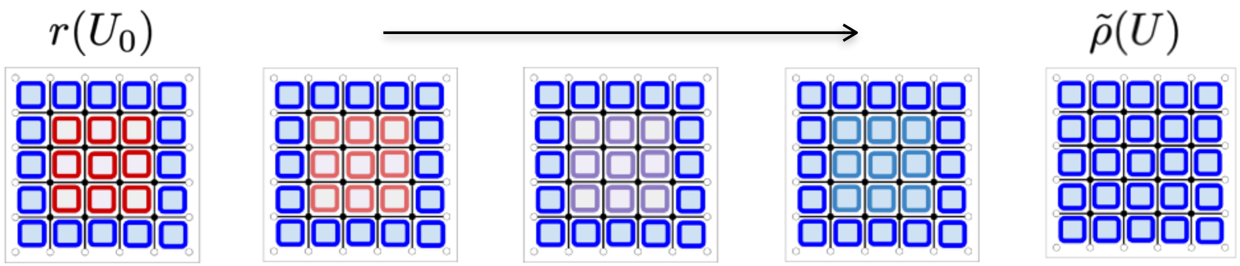
Taken from: M. Luscher, CPC 165 (2005) 199-220

Domain Decomposition of normalizing flow

- update only links/variables inside blocks by creating maps of active links within each block

Training: (one possibility)

- using different boundaries for each sample in the batch
- increase iteration before boundaries updated to 1000



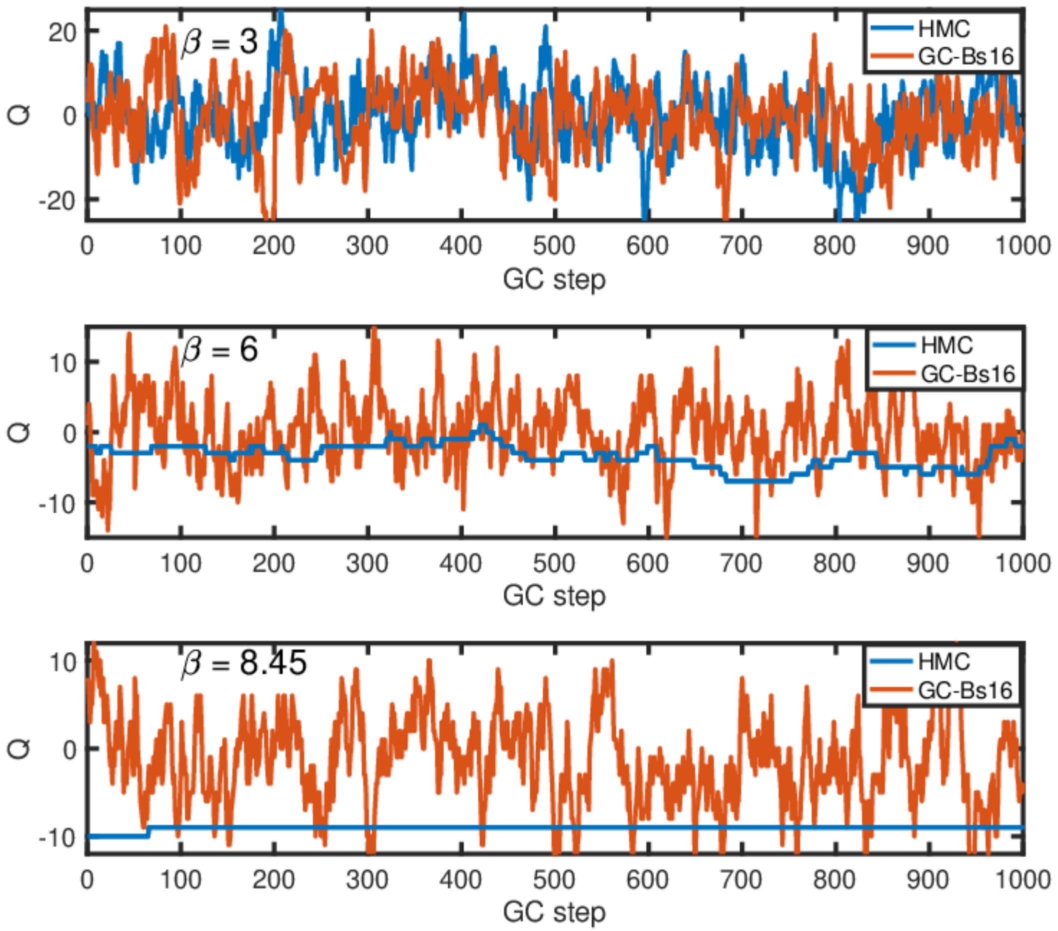
Domain decomposition in the Schwinger Model

Works well in 2D - U(1) gauge model

Train with fixed boundary conditions using L = 8

❖ reduced acceptance rate compare to periodic case (40% to 25%)
❖ Scaling towards large lattice sizes is trivial
❖ Outperform HMC

Domain decomposition works because topological charges are localised



Step towards still tricky 4D - SU(3)

- Block size has to be > 0.4 fm to change topology

For a=0.05 fm, its requires a block of length Lb = 8
- This is currently out of range
  - 6*8*(8)^4 = 200k dof

New approaches needed:
- include additional symmetries
- New training approaches
- Adapt flow maps … there is a lot of room for improvements

Table of Content

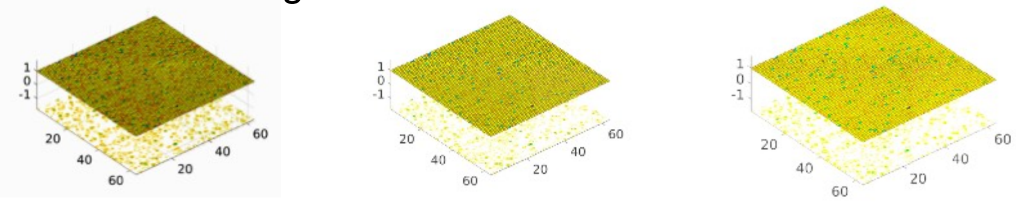
BERGISCHE UNIVERSITÄT WUPPERTAL

FOR 5269

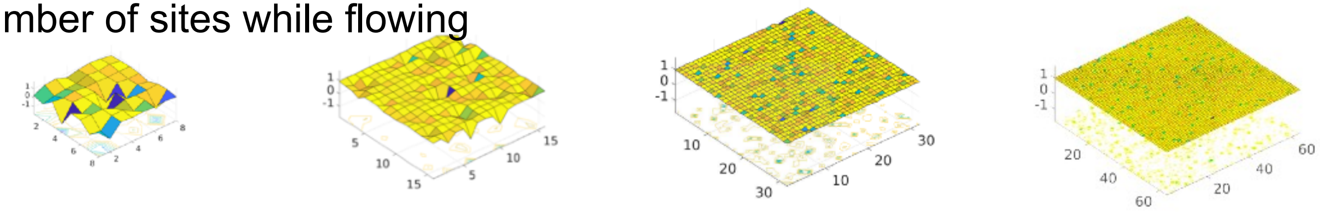
Fine graining flows

Standard Map: Keep L/a fixed
❖ Physical lattice size is decreasing

Flow maps:



Natural Projection: Keep physical box size fixed
❖ extend number of sites while flowing

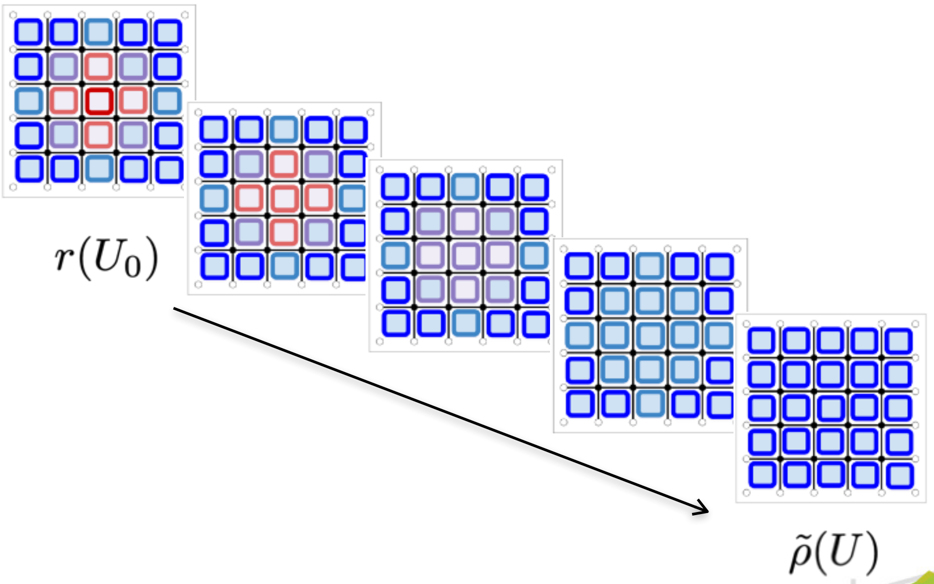


Idea: Effective coarse to fine graining

Introduce a smoothing mapping within a larger lattice
❖ place local defect and smooth out

- like multi-tempering approaches
  - successfully applied in 4D-SU(N)

C. Bonati et al., PRD 99, 054503 (2019)

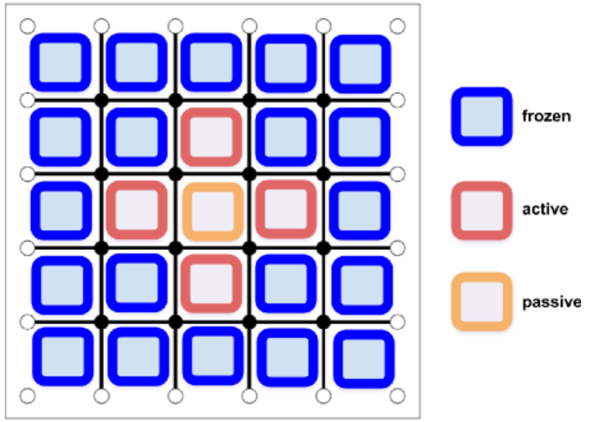


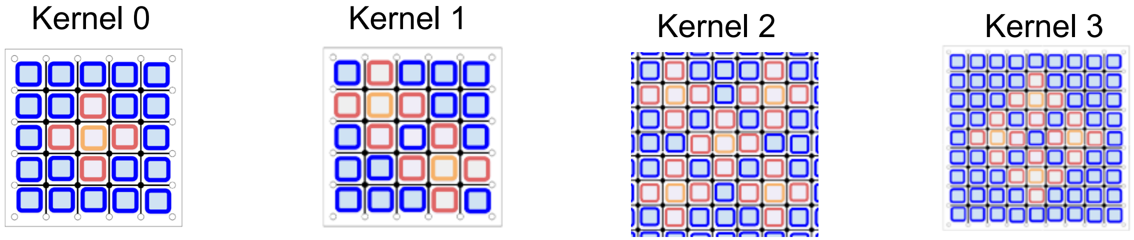
Here: use local flow transformations
❖ needs adjustments/modifications
  ✤ Maps: localization of updates
  ✤ Training conditions / loss function
✤ Train for topological tunneling

FOR 5269

Training techniques

Localized update:
Center symmetric update
❖ only randomise all 4 links of the center plaquette
❖ in 2D use a max. compact map
    ✦ active to passive ratio = 4:1



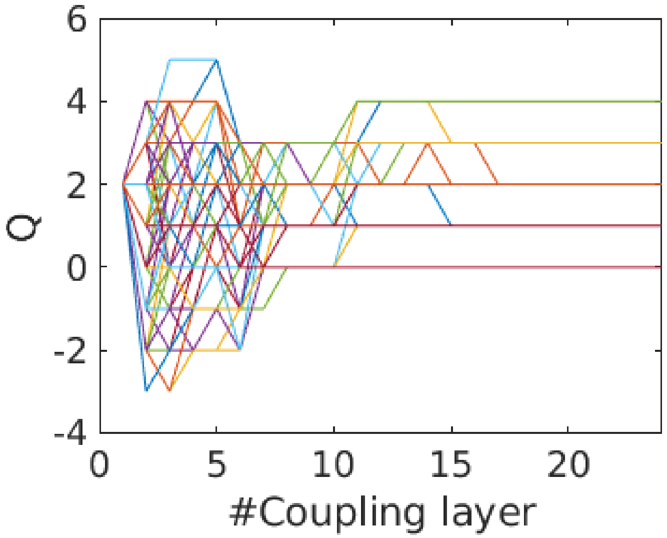


In line with Kanwar et al., Phys.Rev.Lett.125 (2020) 12, 121601

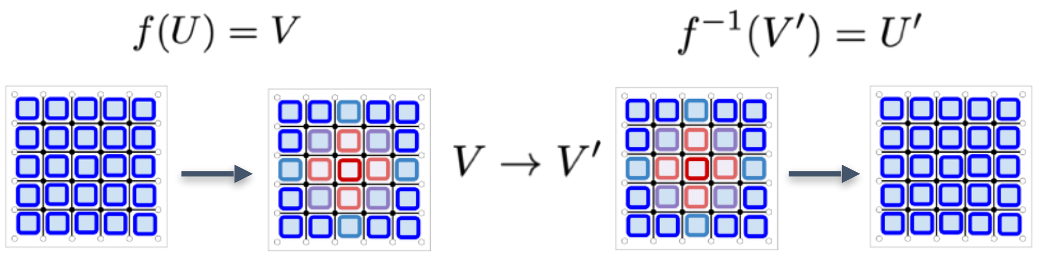
Modification of the loss-function:
Training for topological transitions
❖ by modification of the loss-function

$$L = \ln(\tilde{\rho}(U')/\rho(U')) \cdot |Q(U') - Q(U)|$$

Train transitions using only four uniformed links
❖ Correlations need to be smeared out
❖ otherwise fancy plaquette update



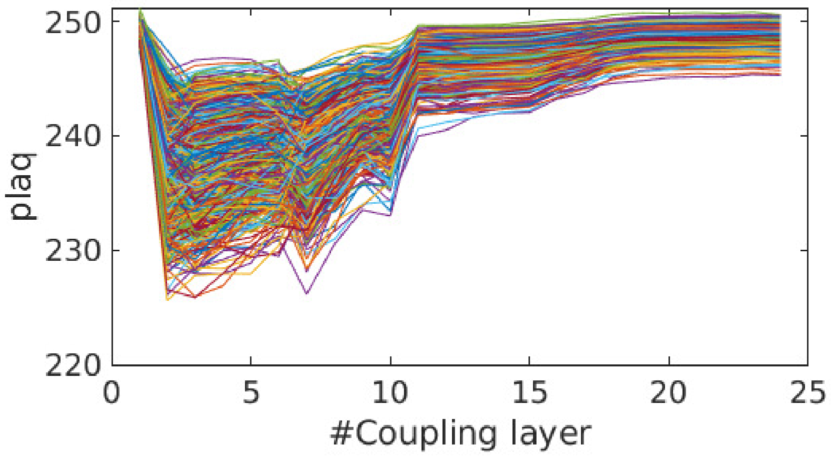
Grinding the fine graining



Graining needs change of update procedure:
Requires back transformation before the update

- Currently: for training fix backward transformation and update occasionally
- Allows for new loss-function
  - Works in combination with topo. loss and fixed backward transformation

$$L = \ln\{\rho(U')\} - \ln\{\rho(U)\} + \ln\{\tilde{\rho}_{fixed}(U)\} - \ln\{\tilde{\rho}_{train}(U')\}$$



Grinding training:
Current training setup: relative long training chain

Pre-train on L=8 with pBC

Retrain on L=16 with fixed $\tilde{\rho}$

Build up chain via intermediate loss

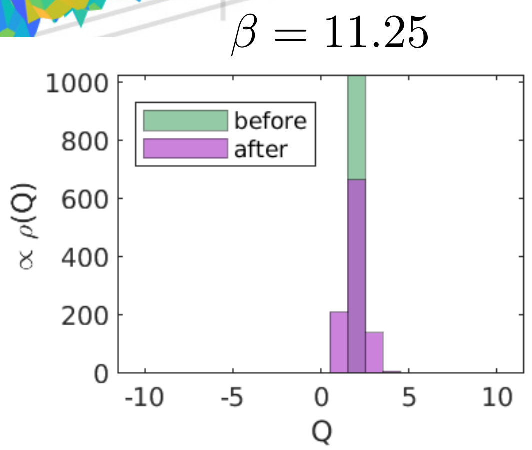
Fine tune with updated $\tilde{\rho}$

Use HMC generated configs

FOR 5269

Results: Tunneling rate

❖ Test, ~30% flipps

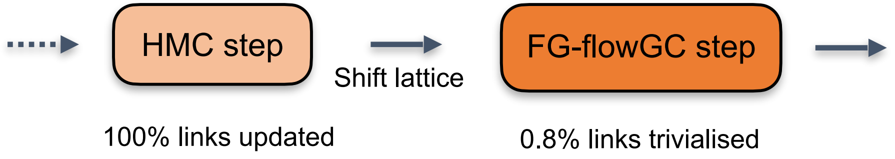


Tunneling rate:

$$T(Q) = \langle |Q_i - Q_{i+1}| \rangle$$

Flow enables simulations beyond beta > 6.0

FG-flowGC:



flowGC:

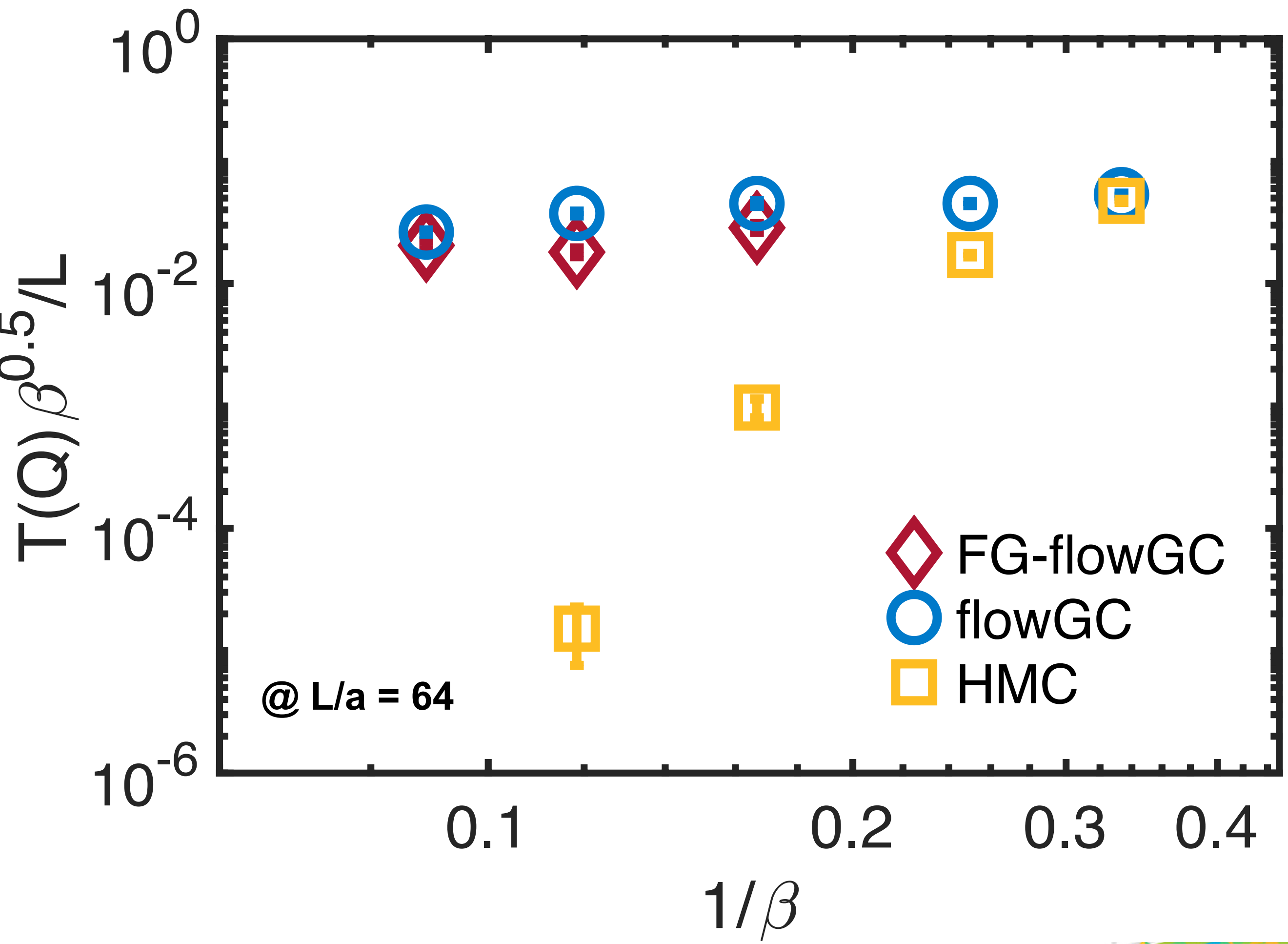


Table of Content

Motivation

- Simulations at the precision frontier
- Monte Carlo simulation

Gauge equivariant/normalizing flows

J. F., arXiv:2201.02216

- Introduction
- Domain Decomposition

Fine graining flows in 2D

J. F., arXiv:2201.02216

- Maps and training
- Tunneling rate

Global corrections with the fermion determinant

- Domain decomposition of fermions
- Towards high acceptances

BERGISCHE UNIVERSITÄT WUPPERTAL

FOR 5269

Global Corrections with Fermions

Recursive Domain Decomposition

Action with fermions:

$$\rho(U) = Z^{-1} \left(\prod_{j}^{N_f} \det D_j(U) \right) e^{-\beta_g(U)}$$

with $\det D(U)$ is a *localised* action

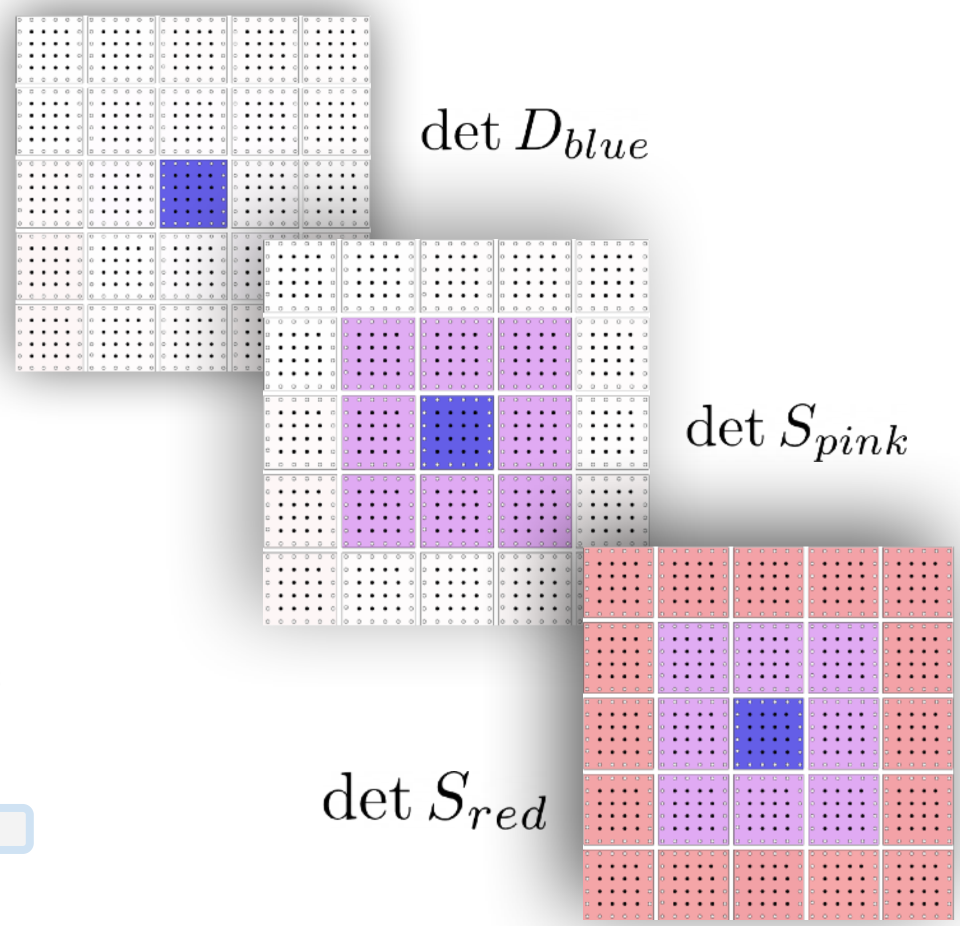
❖ distance interaction decays with

$$cov(x,y) \propto \exp\{-m_{PS}|x-y|\}$$

Idea: using exact decomposition of fermion action:

$$\det D = \det S_{red} \cdot \det S_{pink} \cdot \det D_{blue}$$

effective long range decomposition of the fermion determinant



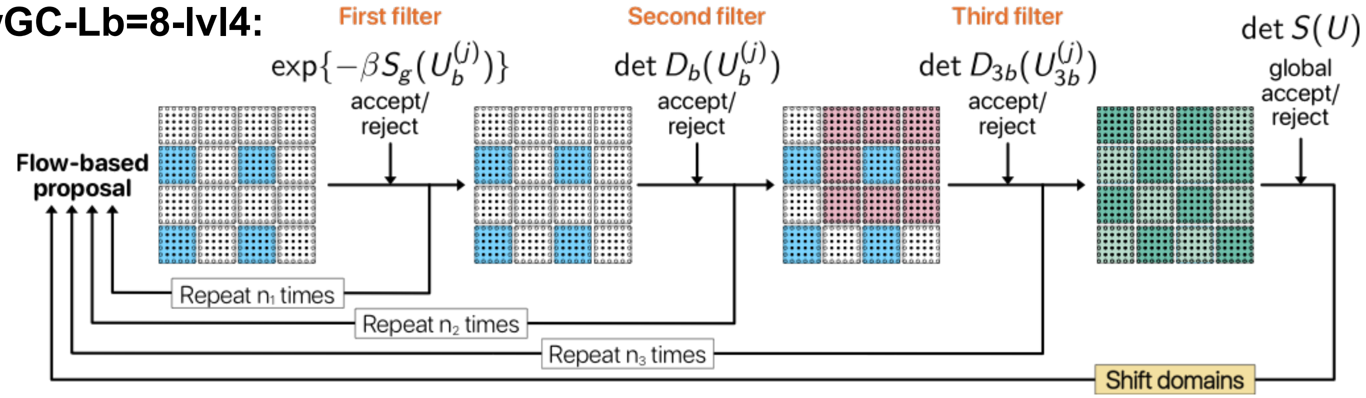
M. Luscher, CPC 165 (2005) 199-220
J.F. et al., CPC 184 (2013) 1522-1534
M. Cè et al., Phys.Rev.D 93 (2016) 9, 094507
M. Cè et al., Phys.Rev.D 95 (2017) 3, 034503

Correction steps With Fermions

Fermion Corrections via hierarchical Filter and flow-based pure gauge updates

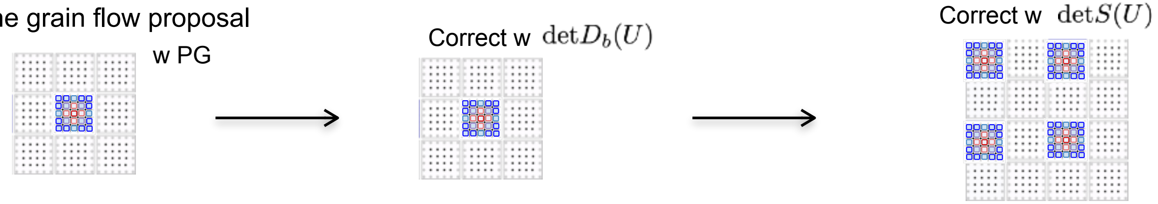
BERGISCHE UNIVERSITÄT WUPPERTAL

flowGC-Lb=8-lvl4:

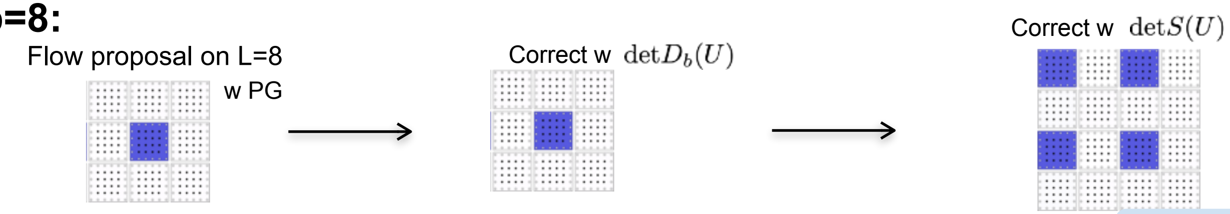


GC-bare:

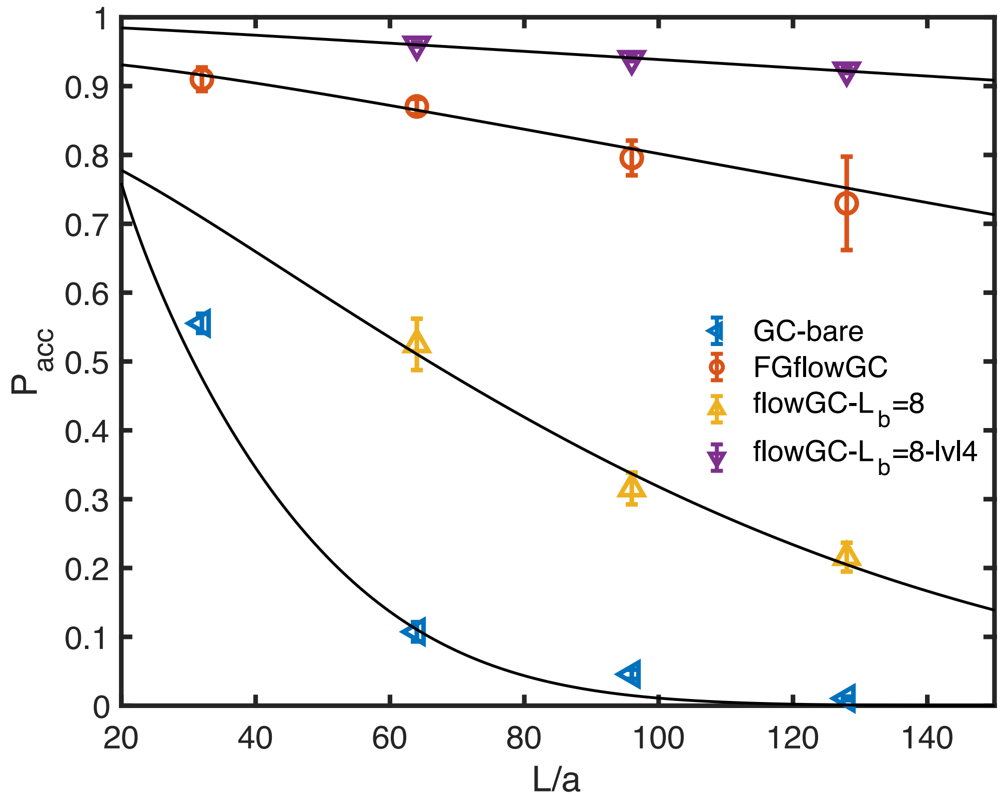
FG-flowGC:



flowGC-Lb=8:



2D - Schwinger Model Global acceptance rate



$\beta = 6.0$ **@ z = 0.2**

N. Christian et al., Nucl.Phys.B 736 (2006)

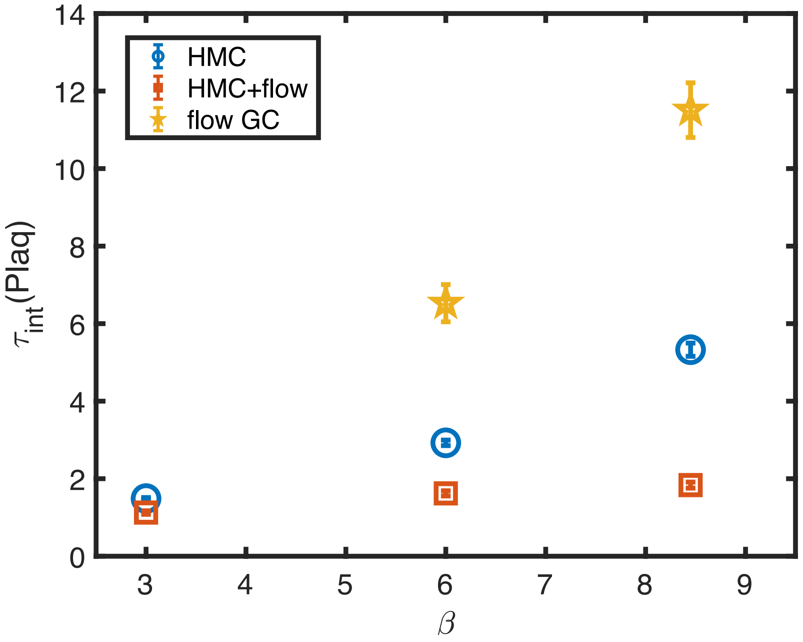
J.F., arXiv:2201.02216

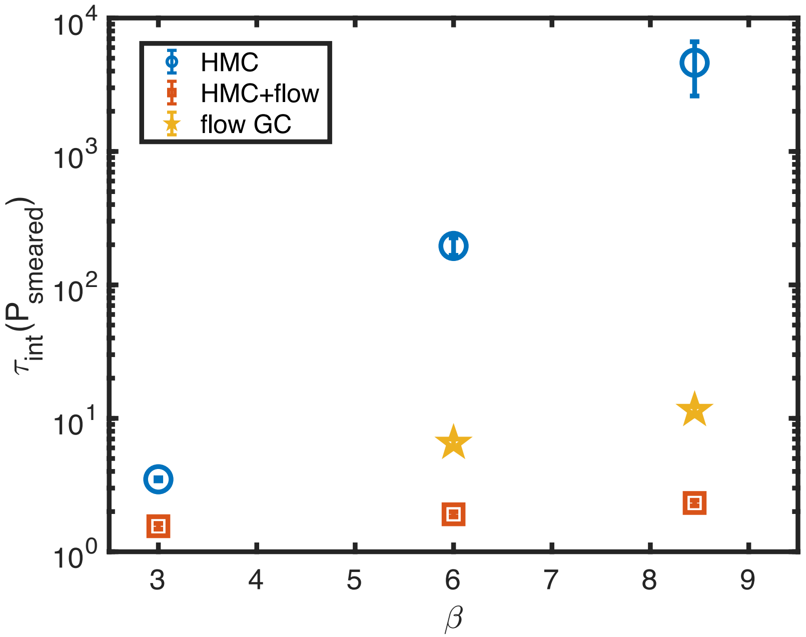
FOR 5269

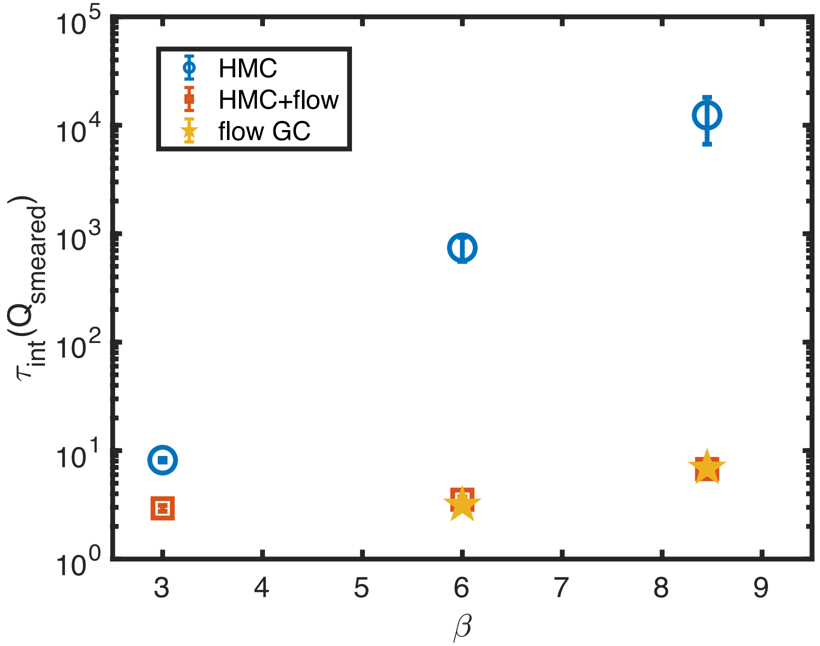
Combination with HMC

High statistic runs (at fixed L=32):
Analyse autocorrelation

Similar to
D. Albandea et al., Eur.Phys.J.C 81 (2021) 10, 873







HMC+flow:
Outperforming other. methods

FOR 5269

Conclusion

BERGISCHE UNIVERSITÄT WUPPERTAL

Addressing scalability

J. F., arXiv:2201.02216

- using domain decomposition
  - Works as long as domain size is larger than *correlation length* of the critical observable
    - Works to sample different topological sectors in gauge models (here U(1))

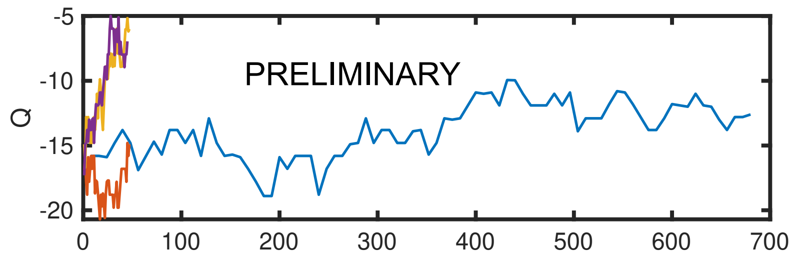
Modification of updates

J. F., arXiv:2402.12176

- fine graining local updates
- Topological transition can be trained in 2D-U(1)

Next steps towards 4D SU(3)

- Complexity increases, more degrees of freedom and topological charge is not an exact integer
- Using flows within multi-tempering approaches looks promising
  - Prove of principle study can move topology at a~0.03 fm



FOR 5269

BERGISCHE UNIVERSITÄT WUPPERTAL

Thank you

FOR 5269

BERGISCHE UNIVERSITÄT WUPPERTAL

Appendix

FOR 5269

Test in the ϕ^4 Model

At 2D parameter set taken from

Korzec et al. Comput.Phys.Commun. 182 (2011)

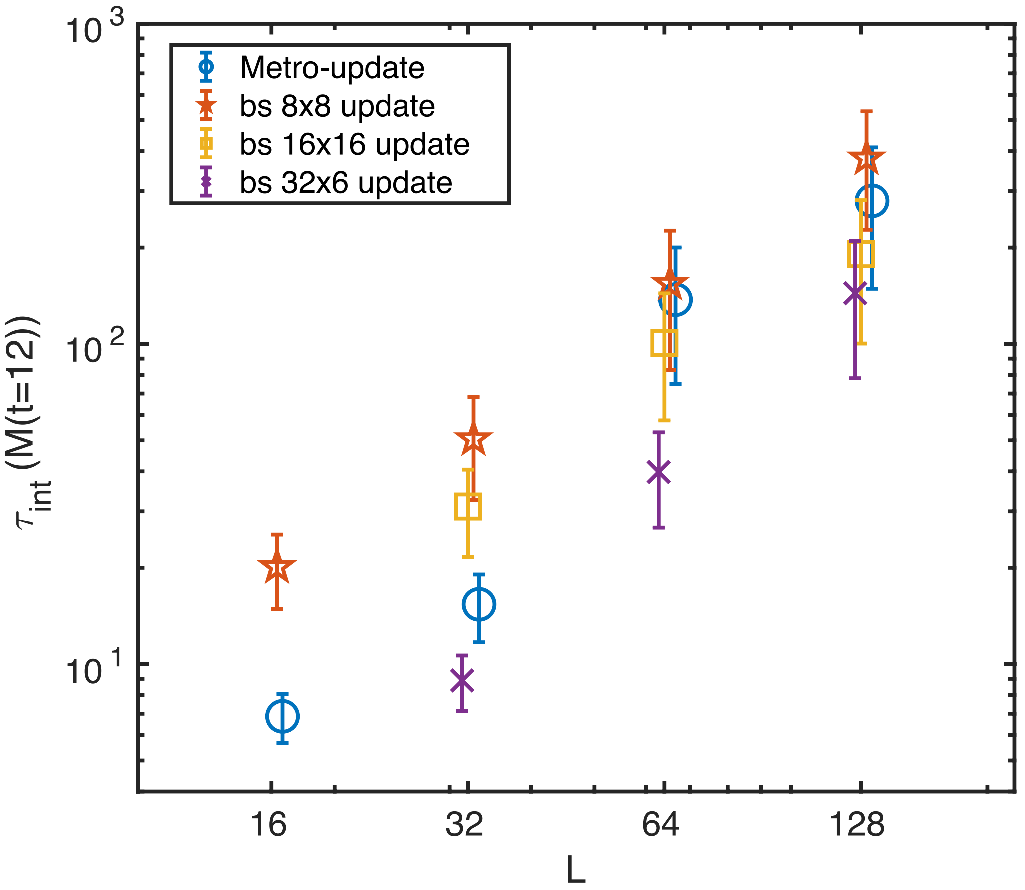
Lets test domain decomposition in the ϕ^4 - model:

$$S(\phi) = \beta \sum_x \phi(x)\phi(x+\hat{\mu}) + \phi^2(x) + \lambda(\phi^2(x) - 1)^2$$

Acceptance rate for flow model become tiny for L> 32
Close to phase transition, model becomes critical.

❖ Order parameter is related to the magnetiziation:

$$M = \frac{1}{V}\sum_x \phi(x)$$



Autocorrelation of magnetisation shows no improvements
if domain size is fixed but lattice size is scaled

FOR 5269

Test in the ϕ^4 Model

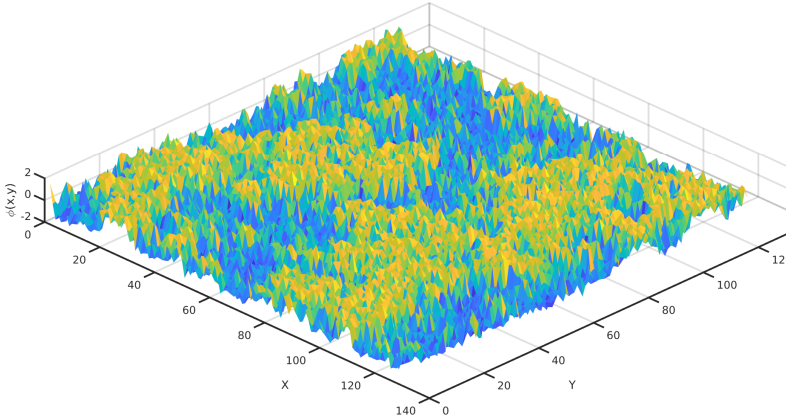
Close to criticality:

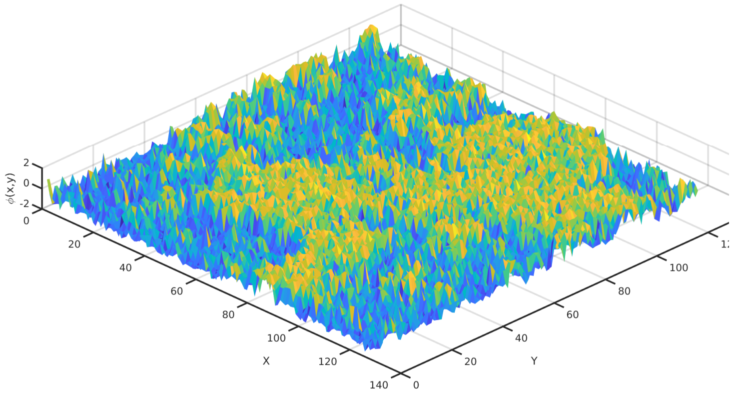
Correlation length increases with lattice size:

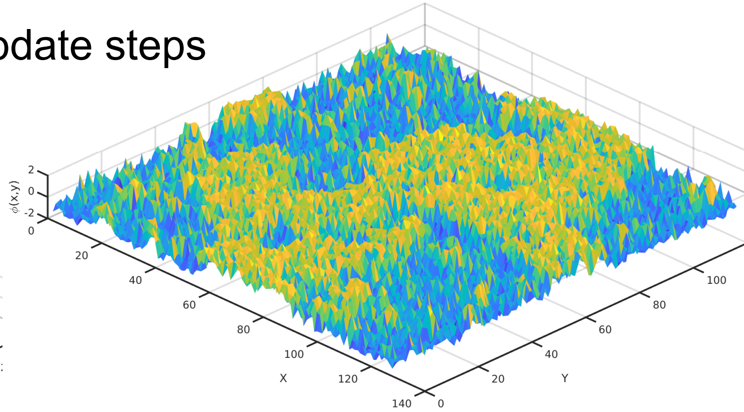
$$\xi \propto L$$

If domain size is smaller < L
the MC-time for de-correlation increases

Here, the magnetisation close to criticality is shown
- using L = 128 and separated by 40k metropolis update steps







FOR 5269

Parallelisation of training

BERGISCHE UNIVERSITÄT WUPPERTAL

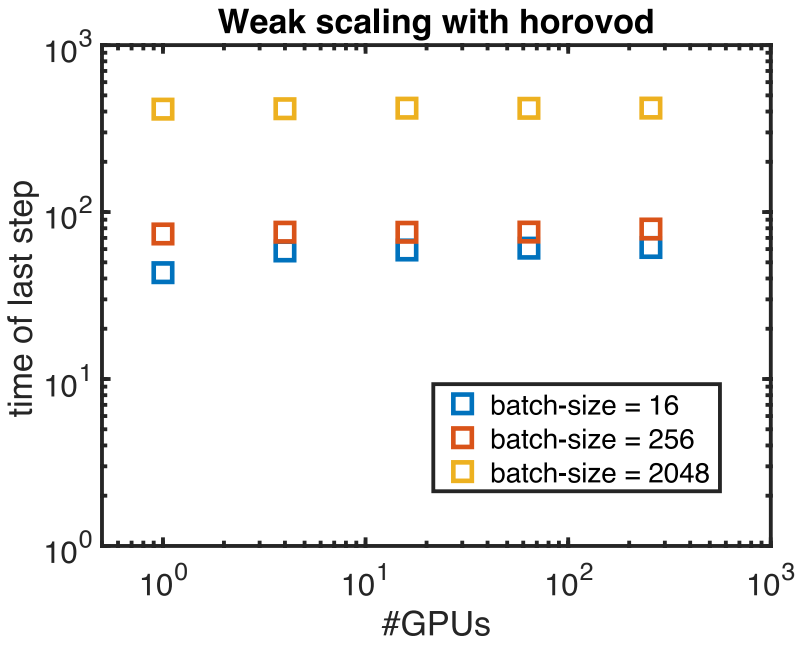
How to scale up:

Exercise with horovod

- Simple to implement but needs fine tuning
- adds new batch to each additional GPU
    - Total batch-size = #GPUs x local batch-size

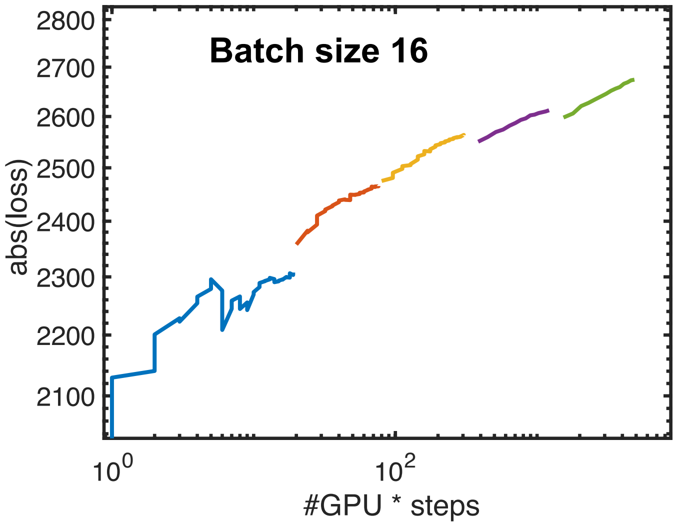
Modifications:

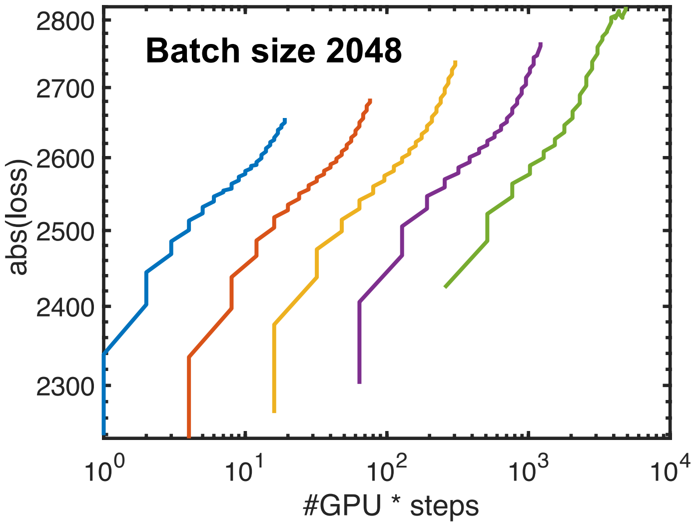
- Switch to double precision
- Use horovod.Adasum
- Use schedular for stepsize decay



Benchmark runs on JUWELS-BOOSTER

- Loosely coupled scales weakly perfect
- For smaller batch-sizes works fine
- For larger batch-sizes convergence deteriorates

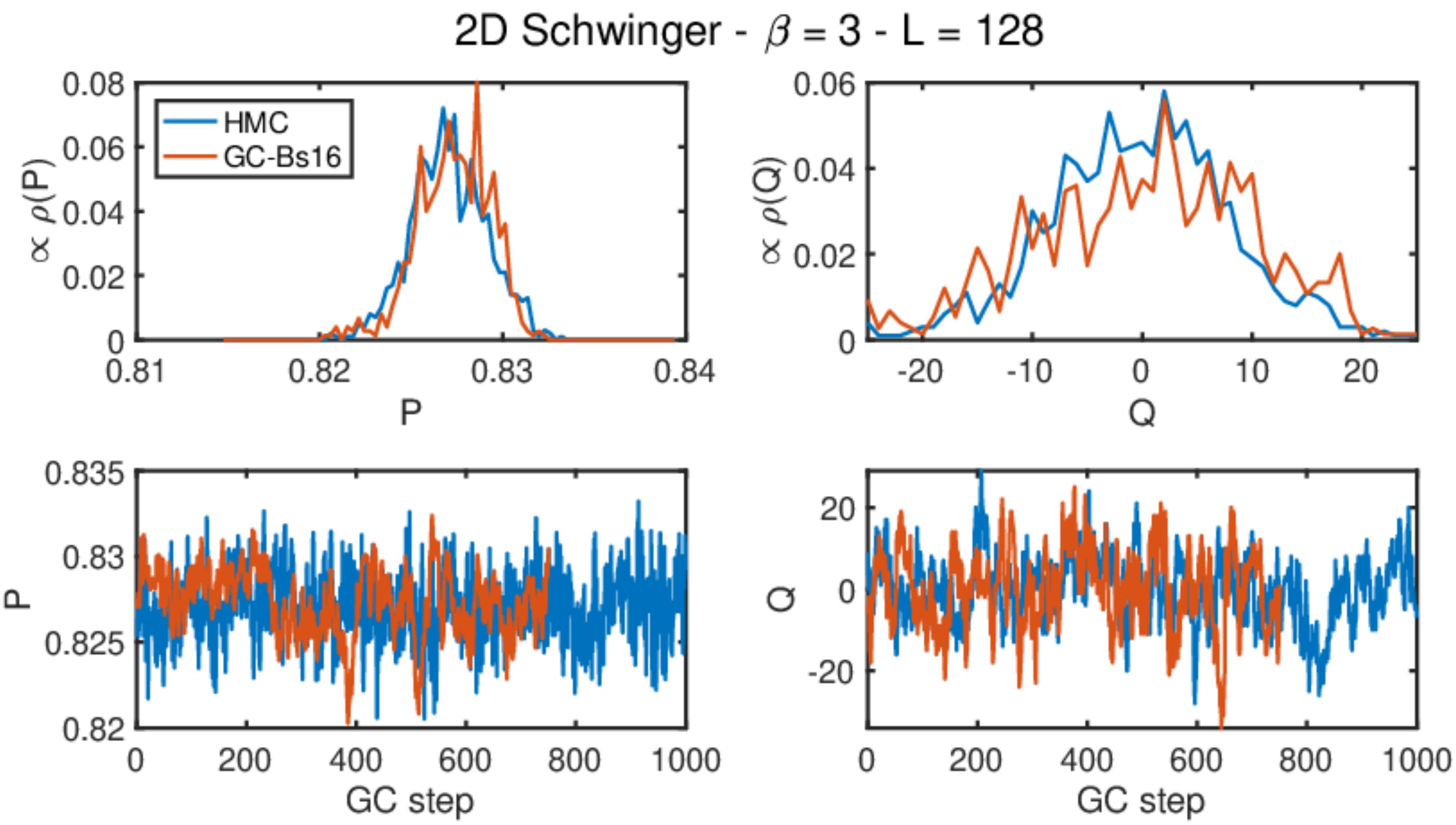




FOR 5269

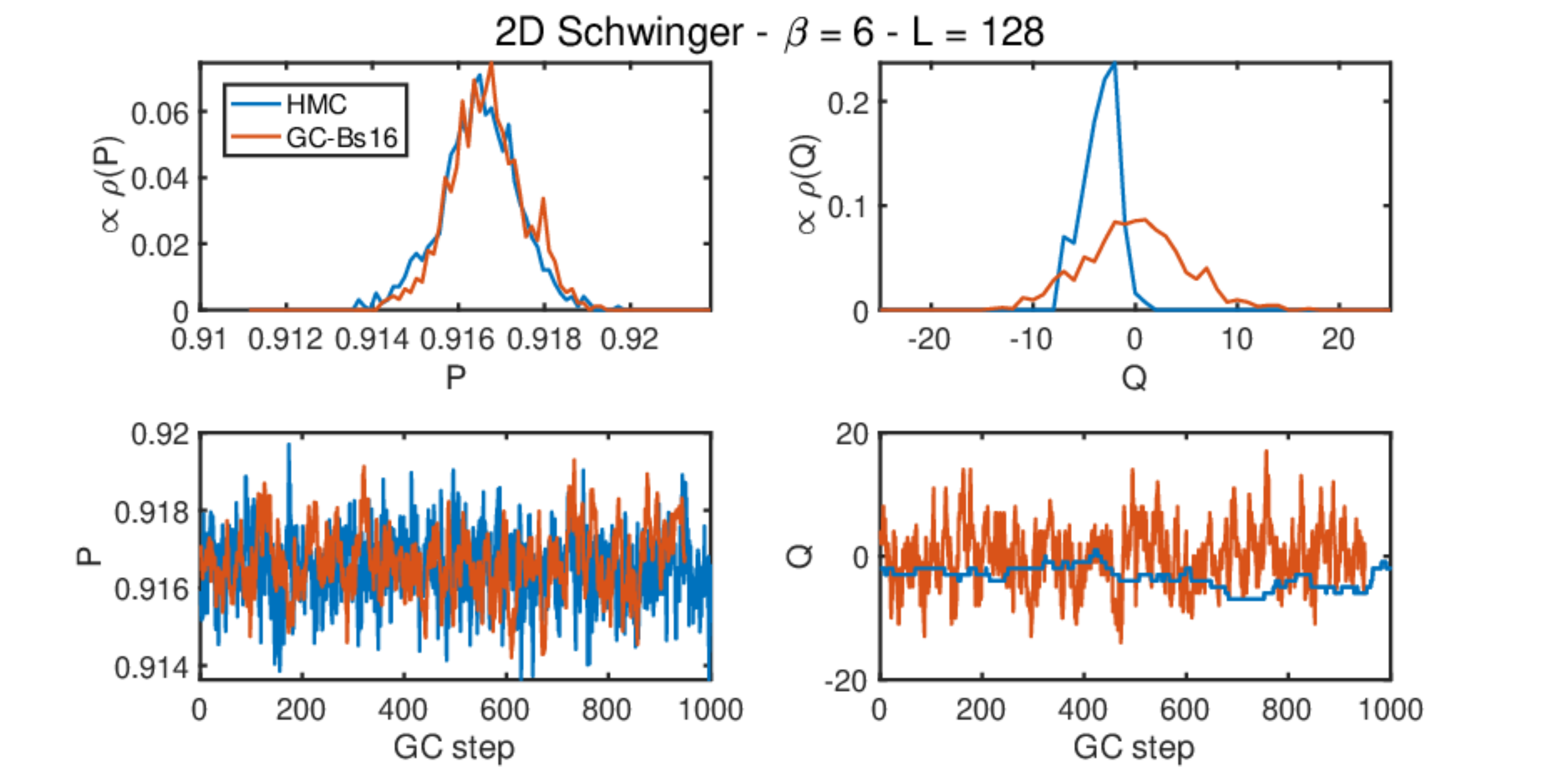
Run statistic

- L = 128
- beta = 3.0
- m = -0.082626



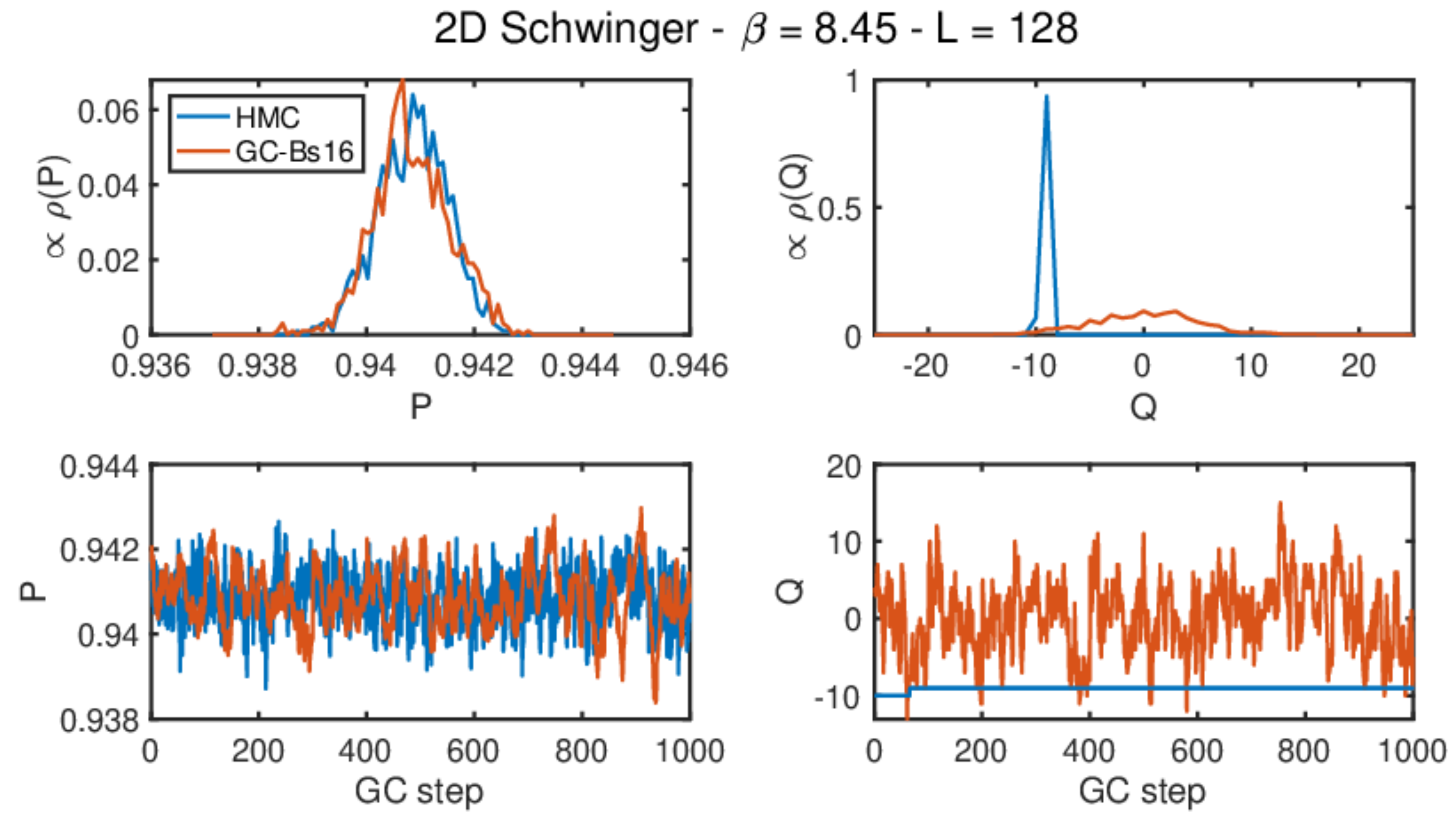
Run statistic

- L = 128
- beta = 6.0
- m = -0.0342



Run statistic

- L = 128
- beta = 8.45
- m = 0.0



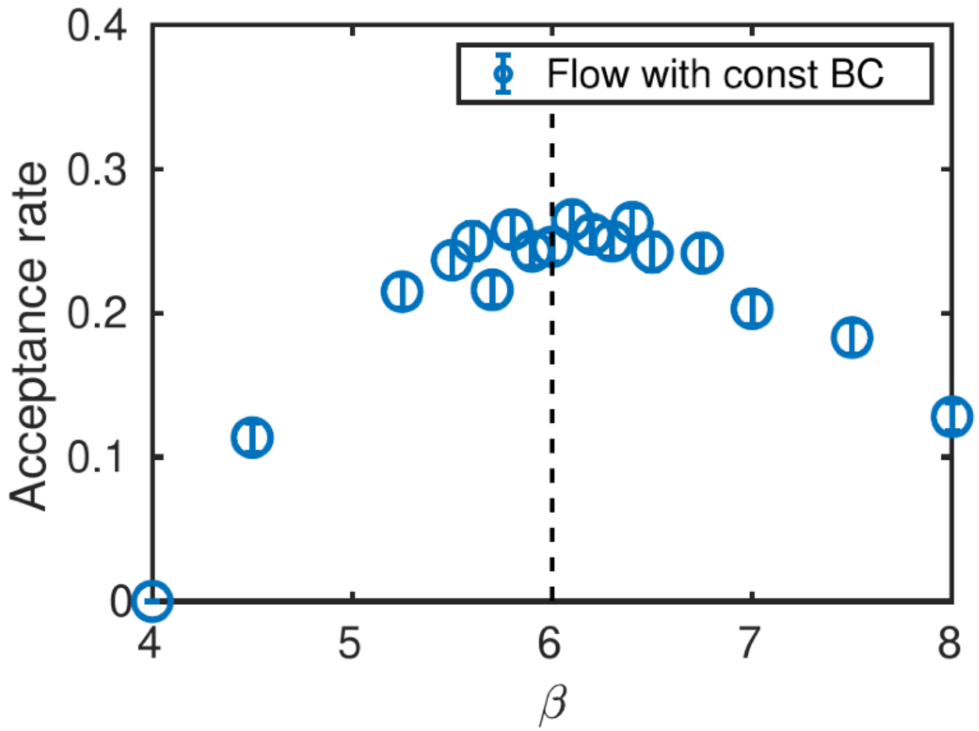
FOR 5269

Training with fix boundaries

Adaptation of training procedure

By:

- Using the periodic trained model to generate boundaries or starting from random and shift lattice after each epoch
- Using different boundaries for each batch with total batch size 4096
- Increase iteration before boundaries updated to 1000
- Using diagonal masks to increase overlap with frozen plaquettes
(faster convergence)



Acceptance rate of fixed boundaries drops down to ~25% with L = 8 (from 50% periodic case)

- due to the ultra locality of gauge action: larger volumes are trivial to generate

FOR 5269

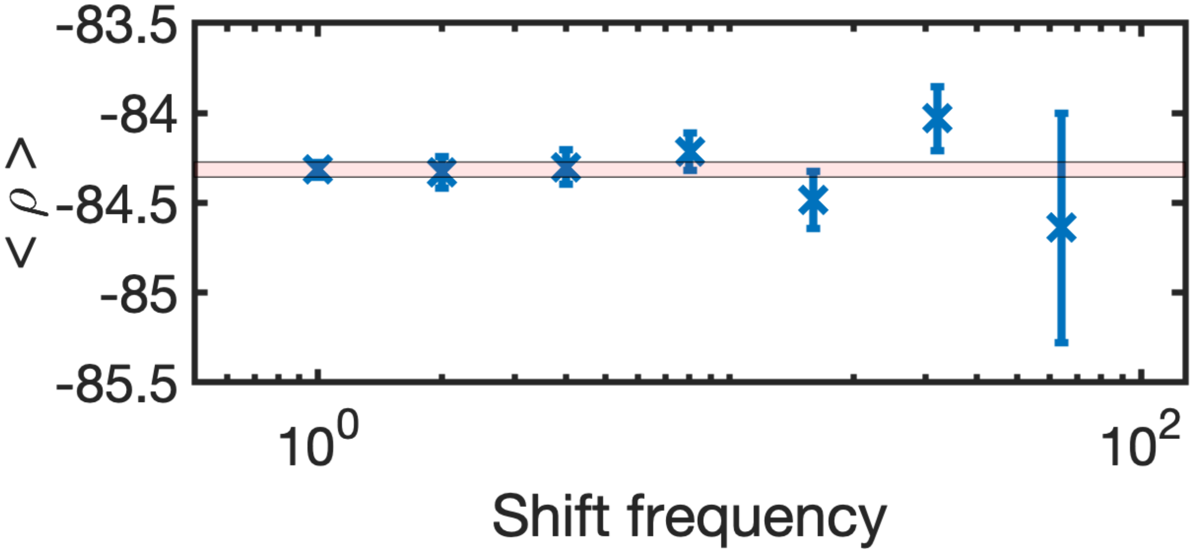
Training with fix boundaries

A priori unknown: Maps under shifts of the lattice

$$T_{\vec{x}} : \vec{x_0} \to \vec{x_0} + \vec{x}$$

Lattice action is invariant but maps *a priori* not

$$\langle \tilde{\rho}(U) \rangle \neq \langle \tilde{\rho}(T_{\vec{x}}(U)) \rangle$$



Idea: check numerical if one can detect deviations

- check if the inverse map maps into the trivial space

$$m^{-1} : T(U) \to T(U_{trivial}) \in \rho_{trivial}?$$

- check if observables are sampled correctly via histograms and via mean values

no numerical indication that shift is violating sampling

- ideally do several iterations of flow updates after a shift

(would eliminate any violation via thermalization)

FOR 5269

Topological sampling in case of the Dirac Index

Topological tunneling requires energy

The Index theorem gives some illustrative insides:

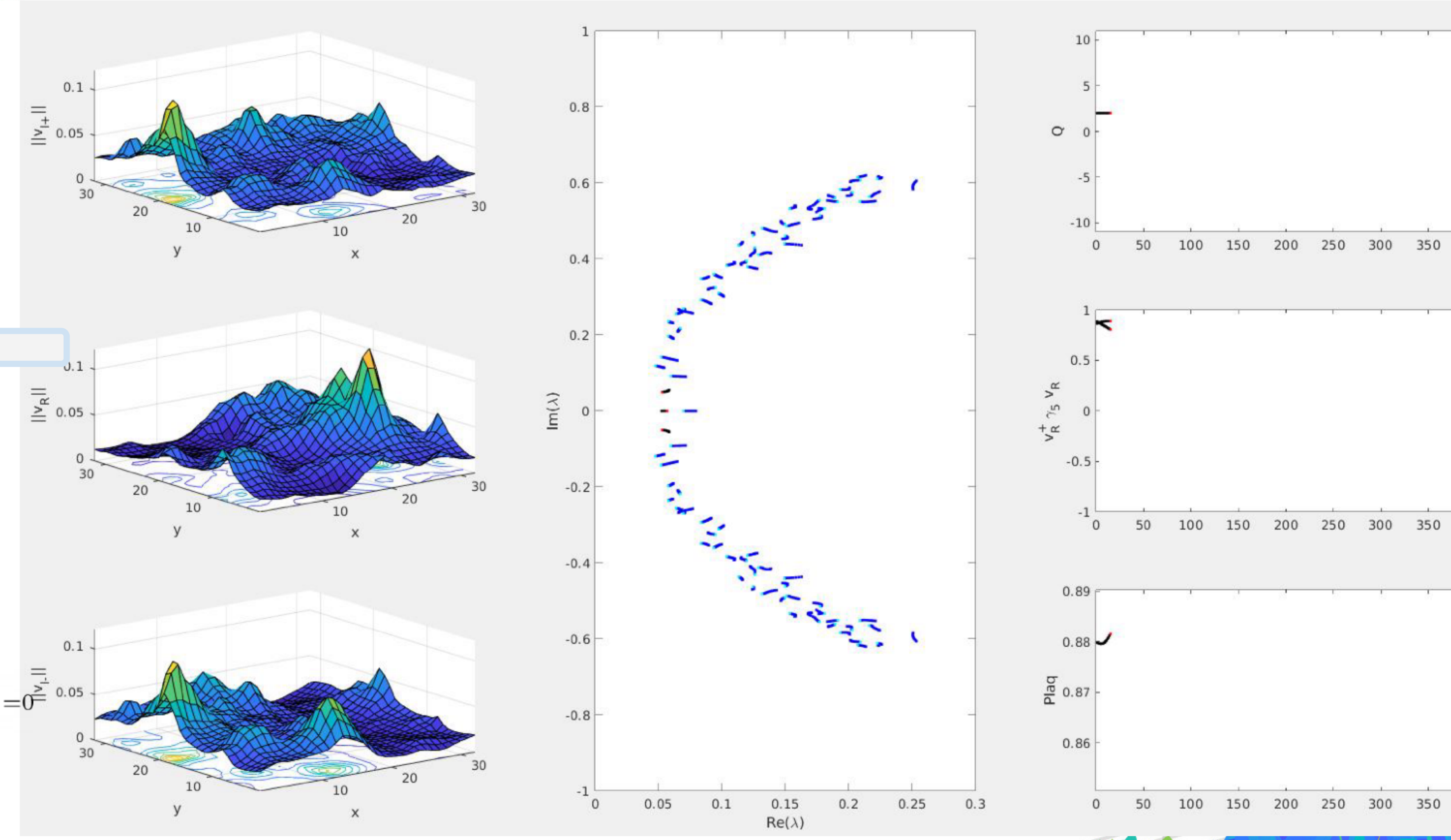
$$N_R - N_L = Q^{geo}$$

Atiyah and Singer, 1963

with the geometric definition:

and $$Q^{geo} = \frac{1}{2\pi}\sum_x \theta_{12}(x)$$

$$\mathrm{Index}(D) = N_R - N_L = \sum_i \chi_i|_{\lambda(D)=0}$$



Modes are localized

❖ **update 8x8 block can flip charge**

Topological sampling in case of the Dirac Index

Topological tunneling requires energy

The Index theorem gives some illustrative insides:

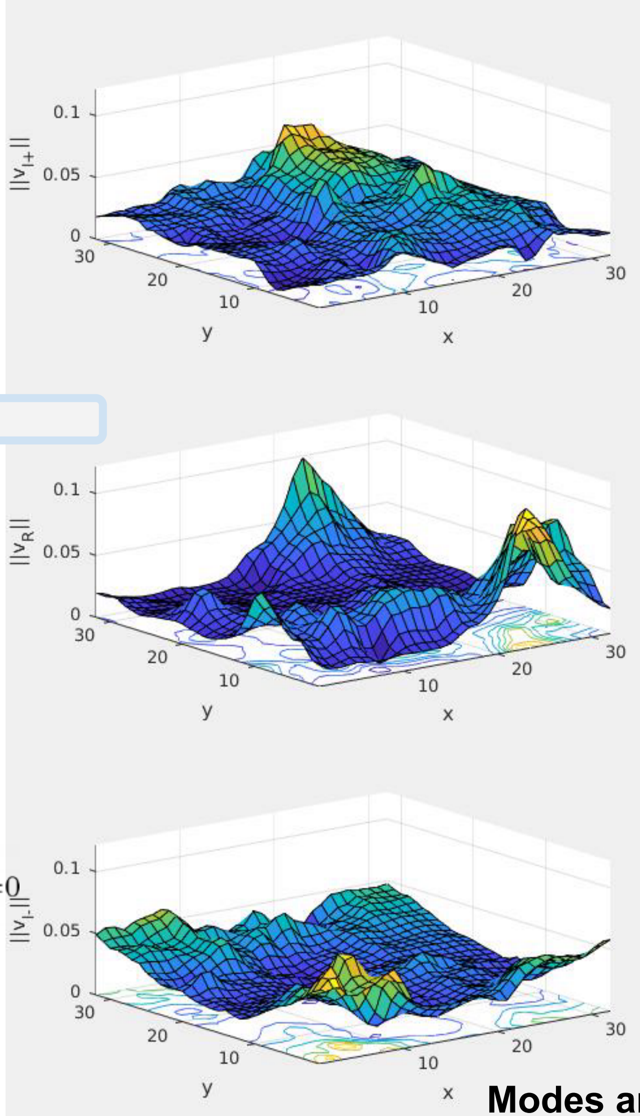
$$N_R - N_L = Q^{geo}$$

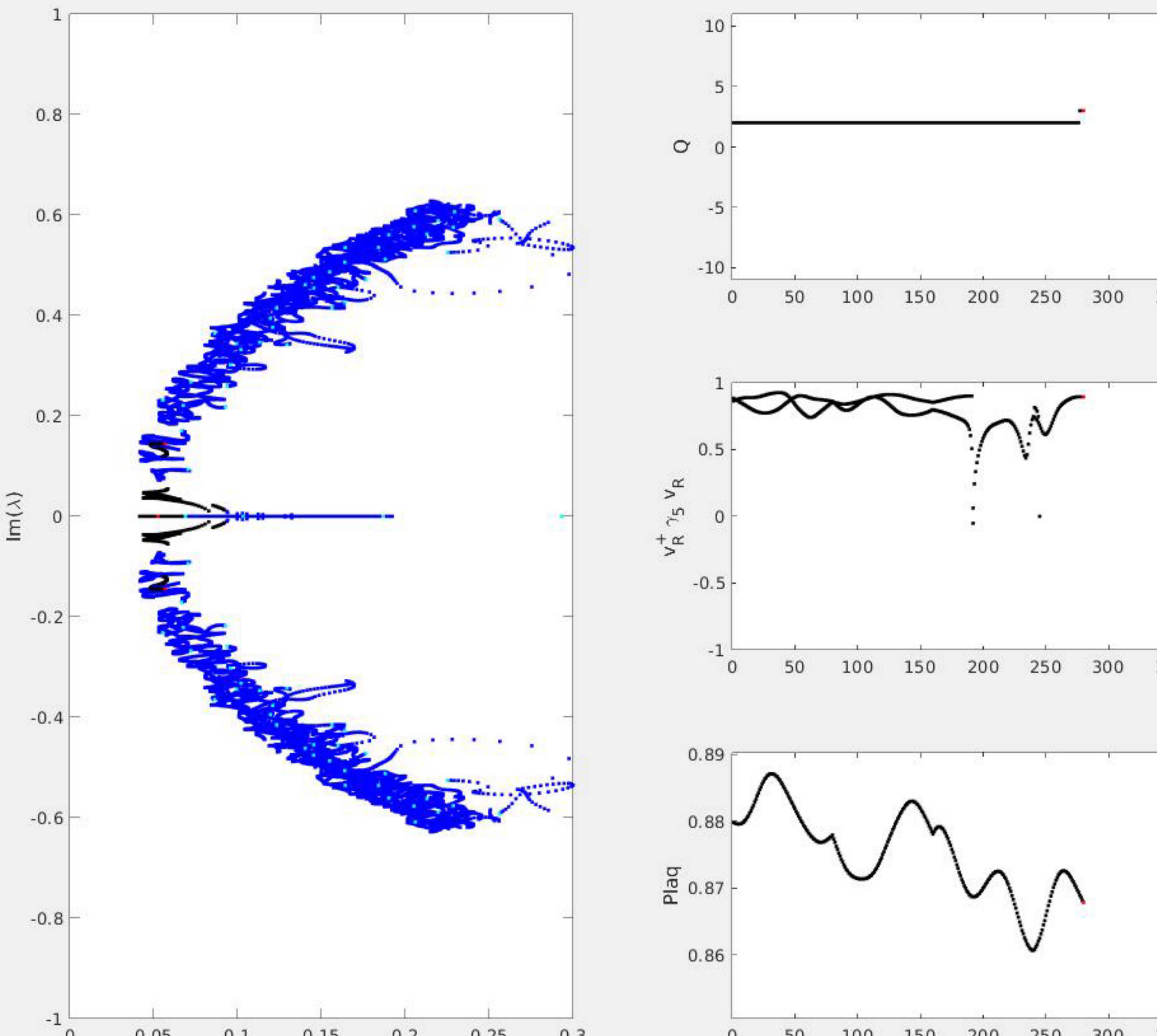
Atiyah and Singer, 1963

with the geometric definition:

and $$Q^{geo} = \frac{1}{2\pi} \sum_x \theta_{12}(x)$$

$$\mathrm{Index}(D) = N_R - N_L = \sum_i \chi_i|_{\lambda(D)=0}$$





Modes are localized

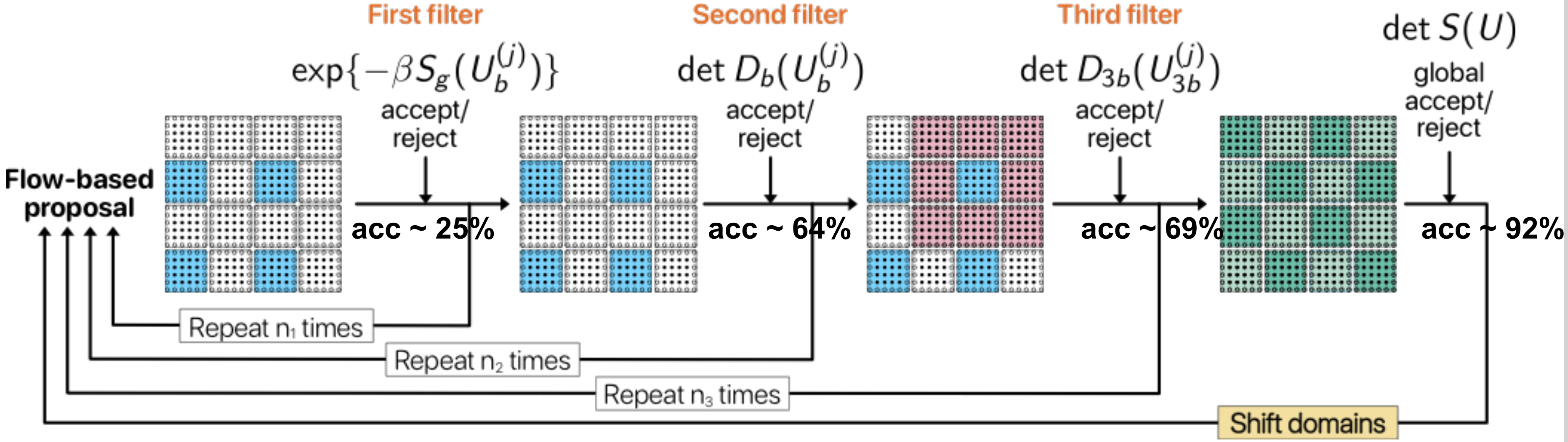
❖ **update 8x8 block can flip charge**

FOR 5269

Towards high acceptance

BERGISCHE UNIVERSITÄT WUPPERTAL

Global Correction Monte Carlo algorithms with equivariant flows:



$$\rho(U) = \det D^2(U)\, e^{-\beta S_g(U)} = e^{-\beta S_g(U)} \prod_b \det D_b^2(U_b) \cdot \det S^2(U)$$

Multilevel hierarchical filter steps with 4 levels

Enhancing acceptance rate by

- within level 1, 2, 3 each active block can be updated independently from each other
- use correlation between actions via parameterization,
  - Using gauge coupling constant $\beta = \beta_0 + \beta_1 + \dots$

β	3.0	6.0	8.45
5 level flowGC with $d = 16$:			
Level 4			
with σ^2	0.0052	0.0369	0.0046
and P_{acc}	0.9713	0.9235	0.9727
$\delta\beta_4^{(3)}$	-2.0037	-2.0182	-2.0087
$\delta\beta_4^{(2)}$	1.0027	1.0061	1.0083
$\delta\beta_4^{(1)}$	-0.0003	0.0008	0.0004
Level 3	$n_1 = 2$		
with σ^2	0.6688	0.6190	0.1546
and P_{acc}	0.6826	0.6940	0.8441
$\delta\beta_3^{(2)}$	-1.1730	-1.3635	-1.3534
$\delta\beta_3^{(1)}$	-0.0006	0.0149	0.0125
Level 2	$n_2 = 4$		
with σ^2	1.4384	0.8325	0.1857
and P_{acc}	0.5487	0.6482	0.8294
$\delta\beta_2^{(1)}$	-0.2482	-0.3082	-0.2863
Level 1	$n_1 = 100$		
with P_{acc}	0.5669	0.2501	0.2794
2 level GC:			
with σ^2	12.3774	9.7119	3.7260
and P_{acc}	0.0786	0.1192	0.3345

J. F., arxiv:2201.02216

FOR 5269

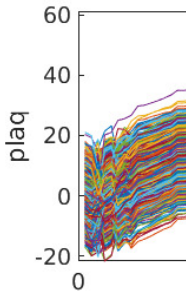
Loss - functions

Build up of coupling layers
- ❖ using intermediate loss-functions
- ❖ Here: for pBC and L=8
- ❖ 64 coupling layers

$$L = \ln\{\tilde{\rho}(U')/\rho(U,\beta)\}$$

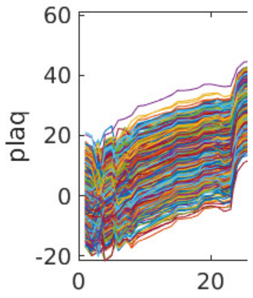
Initial set

$\beta = 0.5$



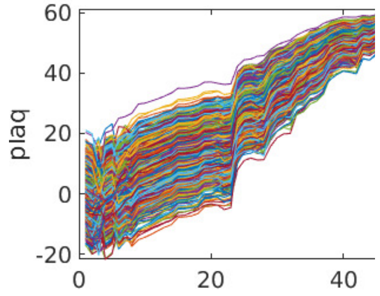
Start after 10 era

$\beta = 2.0$



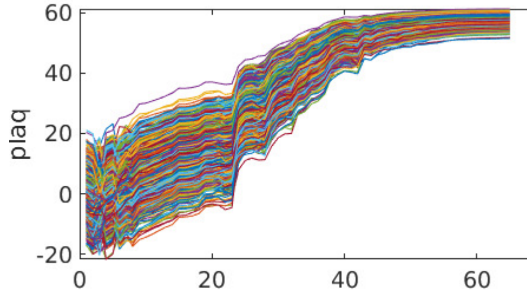
Start after 20 era

$\beta = 5.7$



Start after 40 era

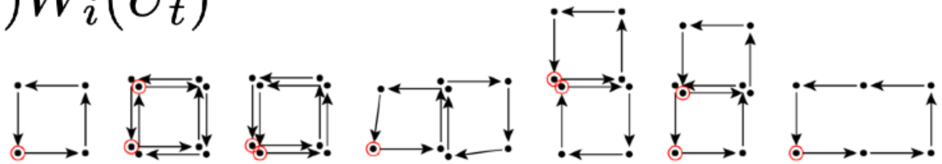
$\beta = 6.0$



- ❖ ideally use action sets from continuous flows (include higher order loops)
- ❖ **Here:** only plaquette action

S. Bacchio et al., 2023

$$\tilde{S} = \sum_i c_i(t) W_i(U_t)$$



FOR 5269