Monte Carlo Methods and Lattice QCD

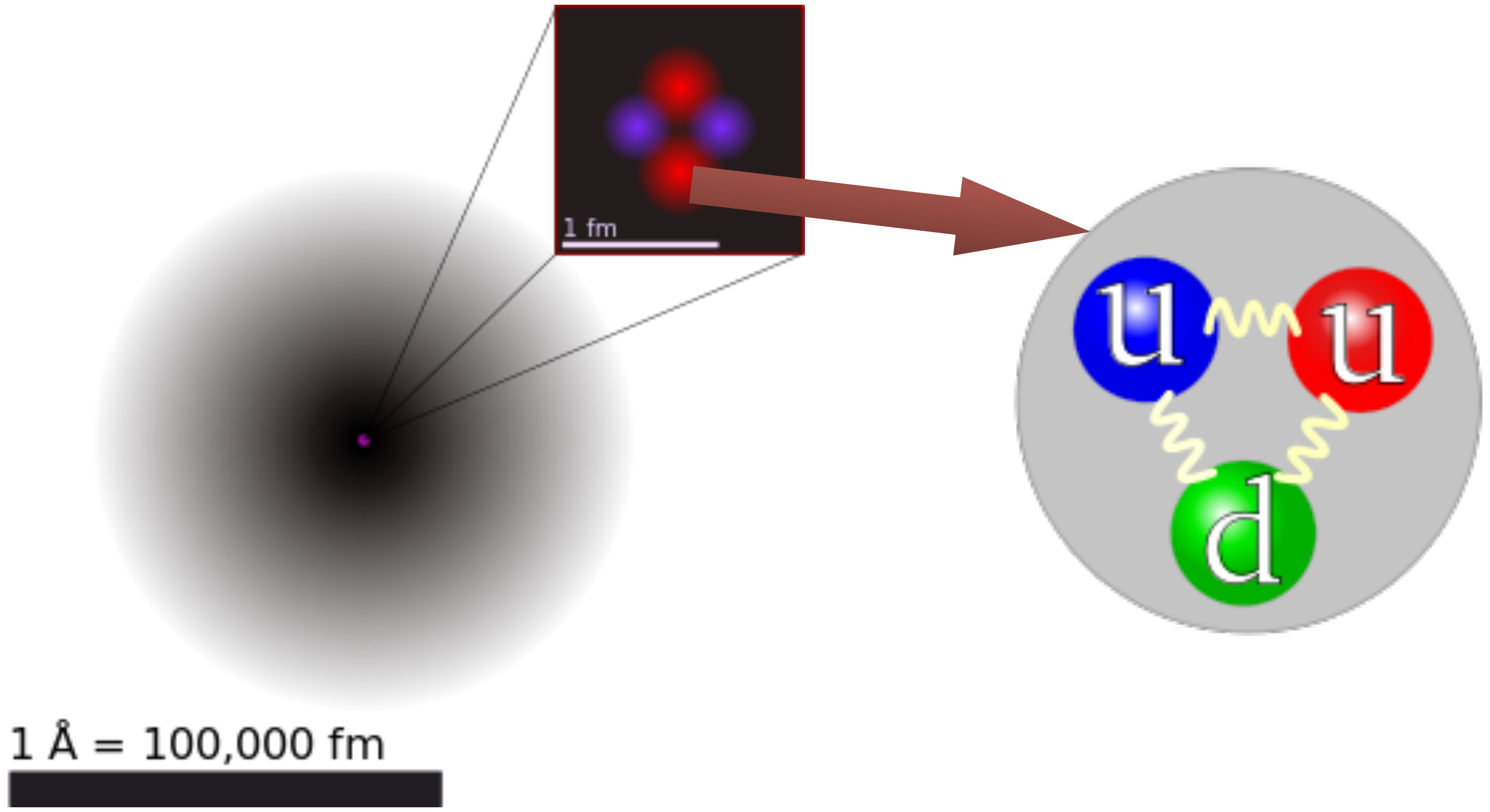
Xiao-Yong Jin
in collaboration with James C Osborn and Sam Foreman

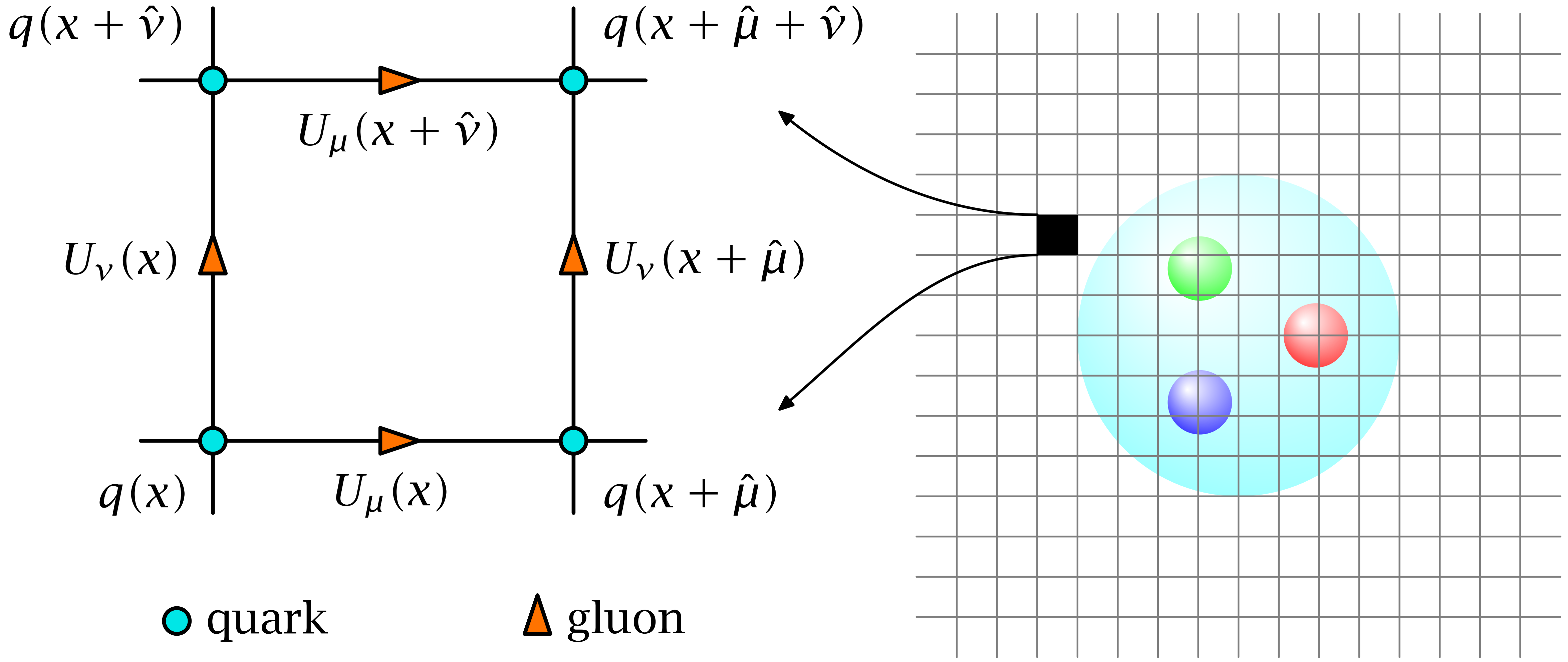
May 18, 2023
Argonne Mini-Workshop on Monte Carlo Methods

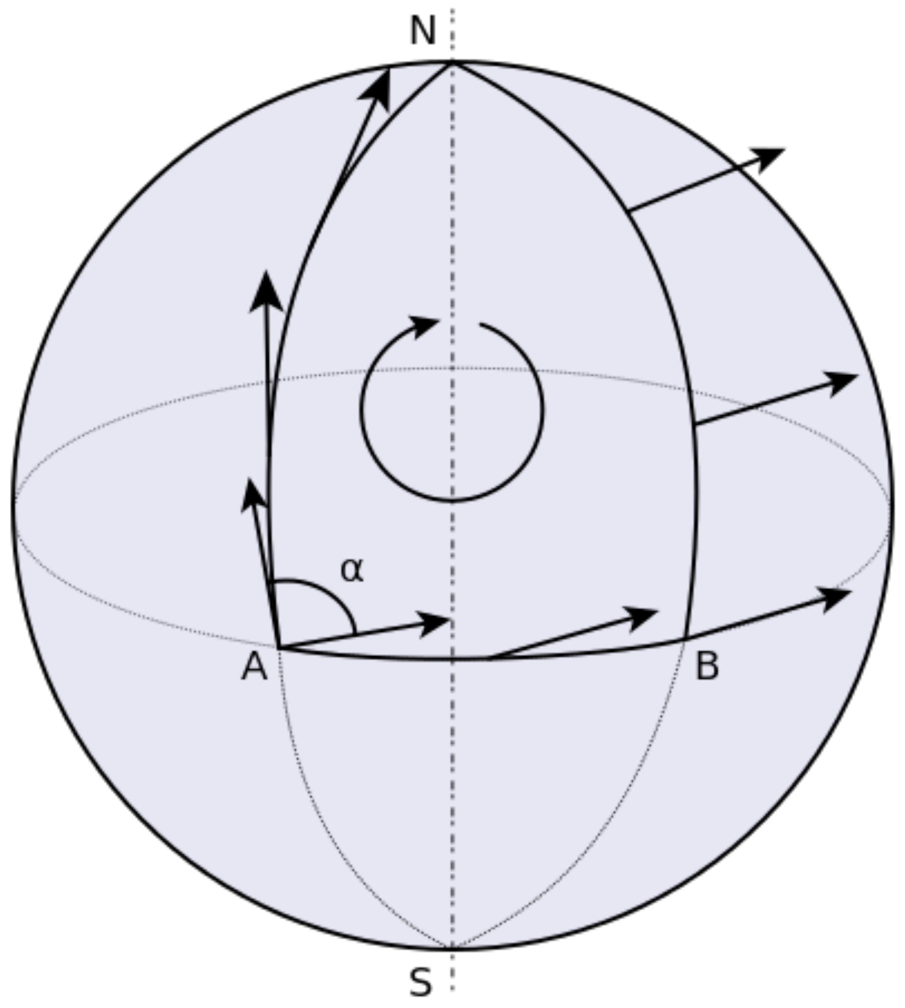
Outline

- Gauge field, fermion, path integral and probability distribution
- Monte Carlo gauge configuration generation
  - HMC - Hybrid Monte Carlo (Duane et al, 1987, and reviewed by Neal, 1993)
    Hamiltonian Monte Carlo (new name by Neal, 2011)
  - Current ML research directions
    - Field transformation, change of variable, parametrized [arXiv:2201.01862]
    - L2HMC, learn to HMC, generalized MD [arXiv:2105.03418, arXiv:2112.01582]
- Monte Carlo in propagator measurement
- Outlook

Gauge field on a spacetime lattice







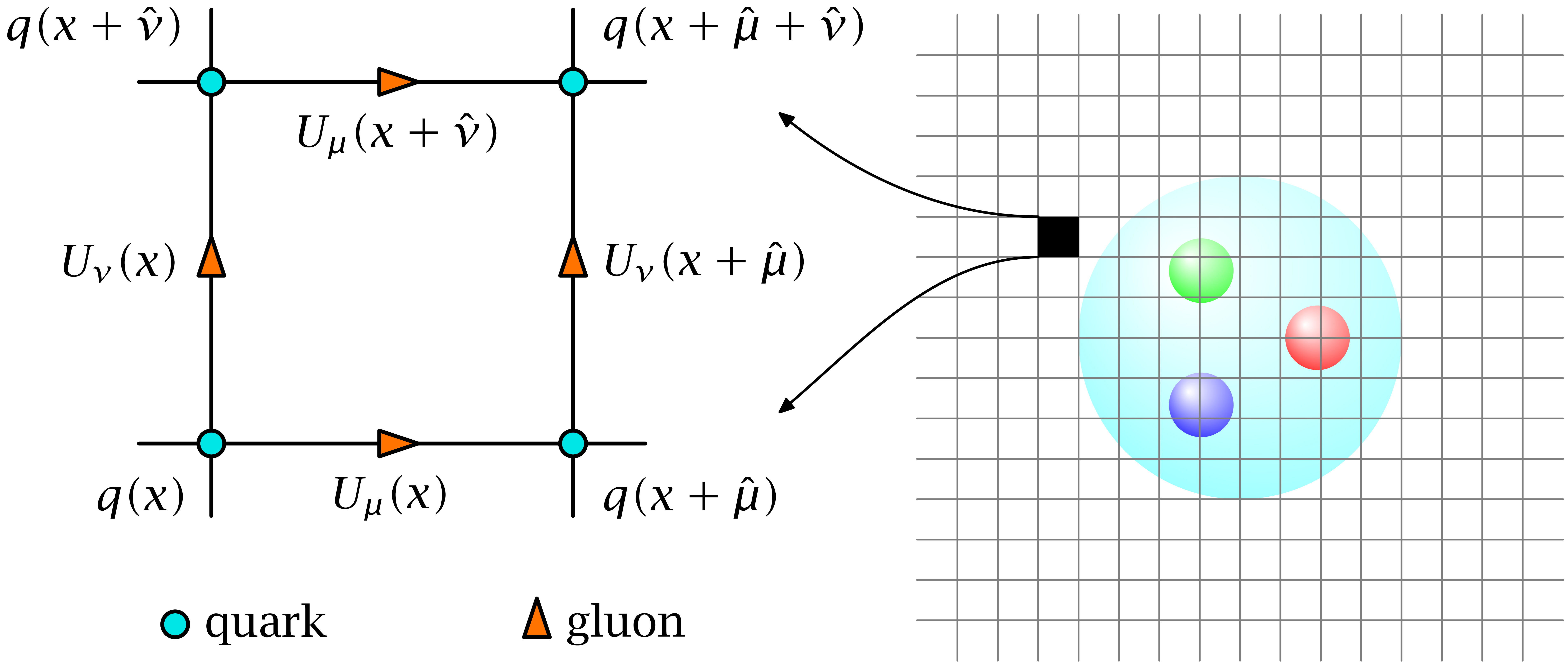
$\Longleftarrow$ example of non-dynamic gauge

Path integral to Monte Carlo

$$\langle\Omega|T\{\phi(x)\phi(y)\}|\Omega\rangle = \lim_{T\to\infty(1-i\epsilon)} \frac{\int \mathcal{D}\phi\, \phi(x)\phi(y) \exp\left[i\int_{-T}^{T} d^4z\, \mathcal{L}\right]}{\int \mathcal{D}\phi\, \exp\left[i\int_{-T}^{T} d^4z\, \mathcal{L}\right]}$$

- Wick rotation $t \to -i\tau$
  - Imaginary time
  - Real action bounded from below
  - Monte Carlo

Lattice gauge theory



$$\langle O(U,q,\bar{q})\rangle = \frac{1}{Z}\int [\mathrm{d}U]\prod_f [\mathrm{d}q_f][\mathrm{d}\bar{q}_f]O(U,q,\bar{q})e^{-S_g[U]-\sum_f \bar{q}_f M[U] q_f} \qquad \int q\mathrm{d}q = 1 \qquad \int \mathrm{d}q = 0$$

Dirac operator, determinant, inverse

***M* local sparse cheap**

$$\langle O(U,q,\bar{q})\rangle = \frac{1}{Z}\int [\mathrm{d}U]\prod_f [\mathrm{d}q_f][\mathrm{d}\bar{q}_f] O(U,q,\bar{q}) e^{-S_\mathrm{g}[U]-\sum_f \bar{q}_f M[U] q_f}$$

$$= \frac{1}{Z}\int [\mathrm{d}U]\prod_f [\mathrm{d}q_f][\mathrm{d}\bar{q}_f]\; O(U,\frac{\partial}{\partial\eta},\frac{\partial}{\partial\bar{\eta}}) e^{-S_\mathrm{g}-\sum_f \bar{q}_f M[U] q_f + \bar{q}_f\eta_f + \bar{\eta}_f q_f}\Big|_{\eta=0,\bar{\eta}=0}$$

$$= \frac{1}{Z}\int [\mathrm{d}U]\prod_f \det\big[M_f\big]\; O(U,\frac{\partial}{\partial\eta},\frac{\partial}{\partial\bar{\eta}}) e^{-S_\mathrm{g}+\sum_f \bar{\eta}_f M_f \eta_f}\Big|_{\eta=0,\bar{\eta}=0}$$

$$= \frac{1}{Z}\int [\mathrm{d}U]\prod_f \det\big[M_f\big] O(U,M^{-1}) e^{-S_\mathrm{g}}$$

$$= \frac{1}{Z}\int [\mathrm{d}U]\prod_f [\mathrm{d}\varphi_f] O(U,M^{-1}) e^{-S_\mathrm{g}-\sum_f \varphi_f^\dagger M_f^{-1}\varphi_f}$$

***M*$^{-1}$ nonlocal expensive**

Dirac operator

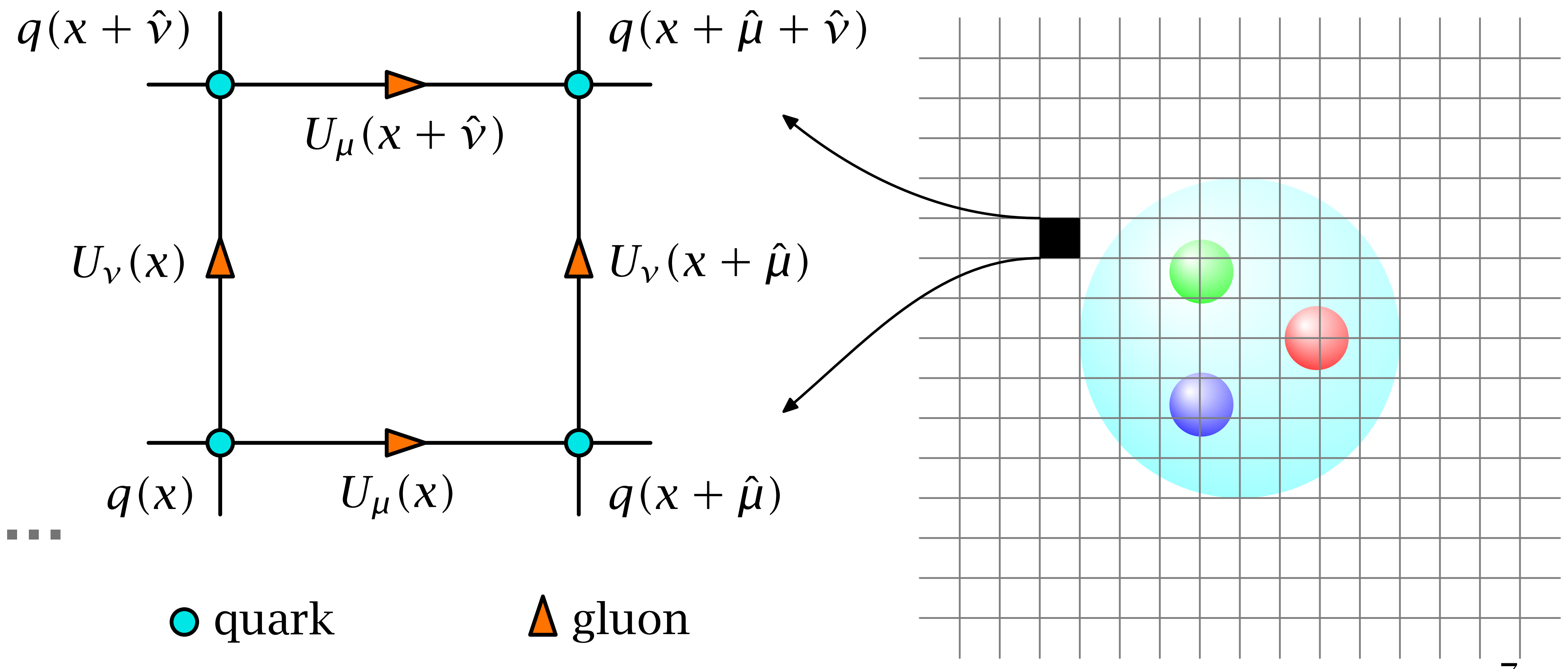
$$S_g = \beta \sum_{x,\mu,\nu} \left(1 - \frac{1}{3} \mathrm{Re\,Tr}\left[U_\mu(x) U_\nu(x+\hat{\mu}) U_\mu^\dagger(x+\hat{\nu}) U_\nu^\dagger(x)\right]\right)$$

$$S_q = \sum_{f,x,y,\mu} \bar{q}_f(y) \left(D_{\mu,yx}[U]\gamma_\mu + m_f\right) q_f(x)$$

$$D_\mu^{\mathrm{Naive}} q(x) = \frac{1}{2}\left(U_\mu(x) q(x+\hat{\mu}) - U_\mu^\dagger(x-\hat{\mu}) q(x-\hat{\mu})\right)$$

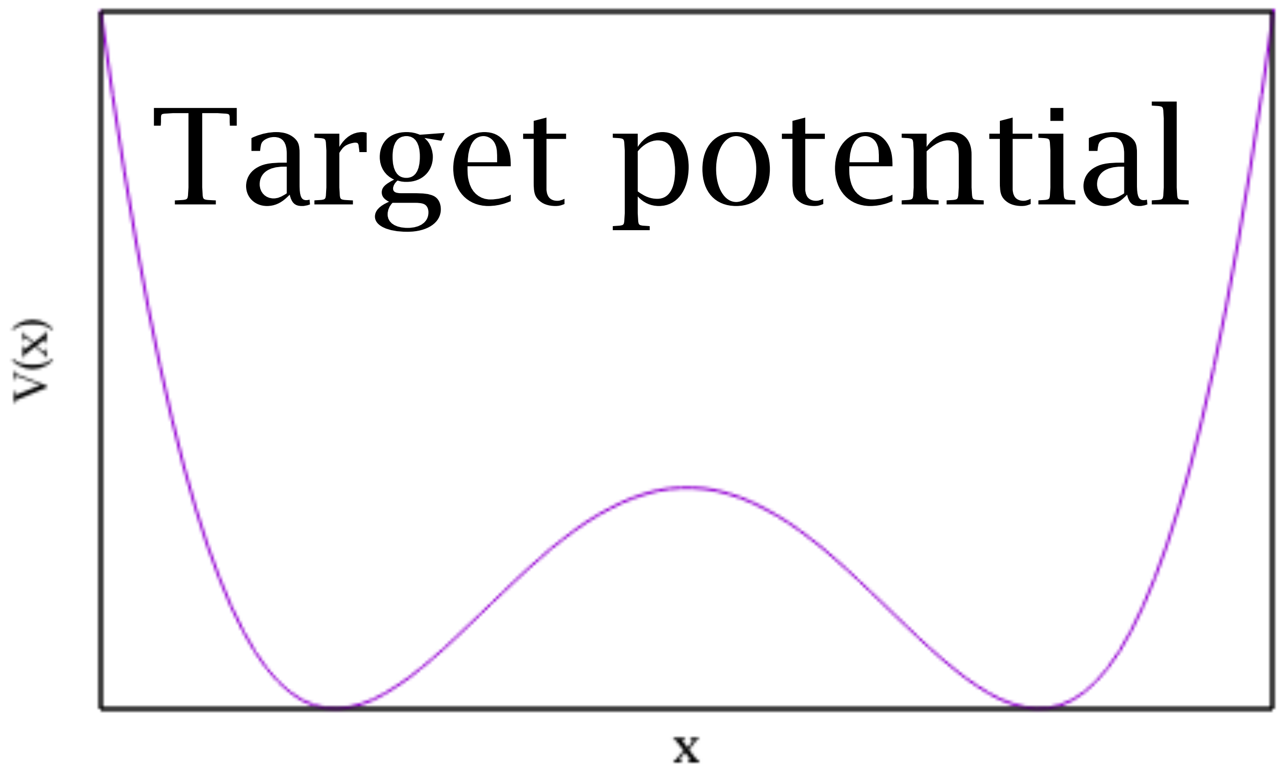
- Naive Dirac operator leads to doubling of momentum poles per dimension

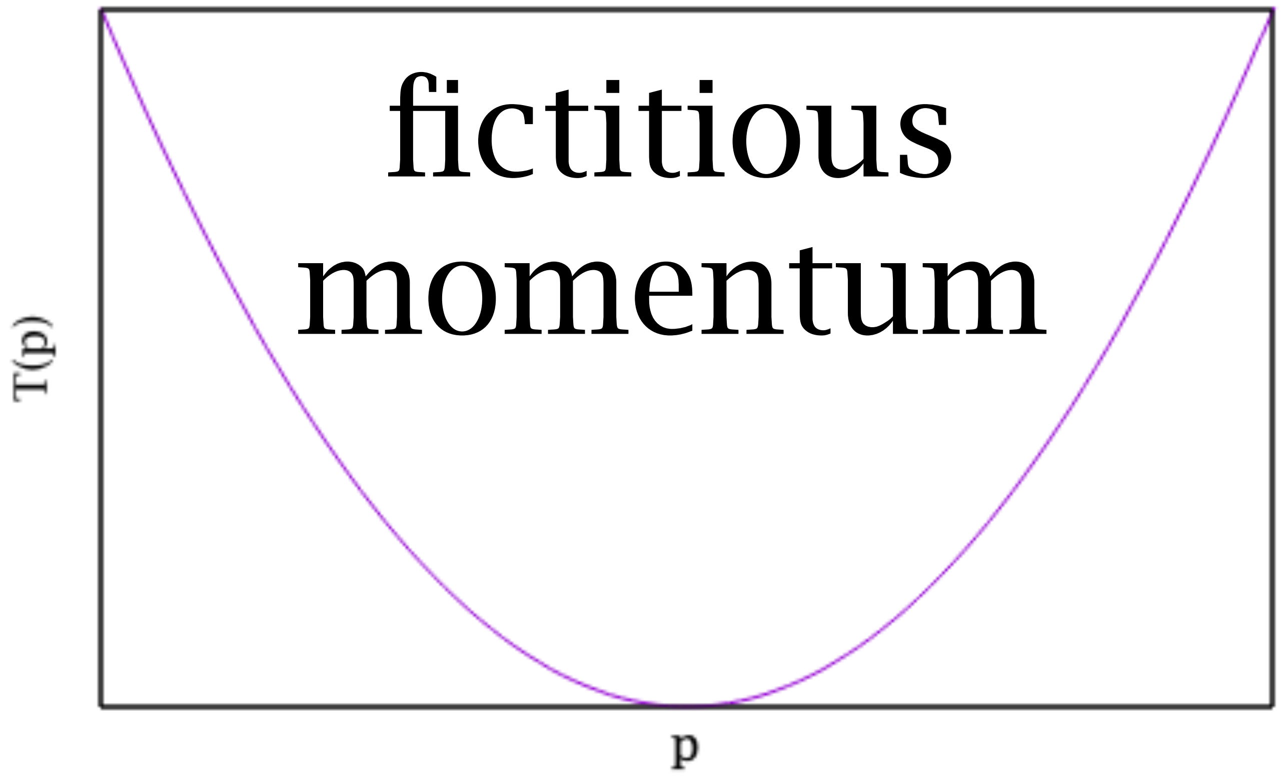
- Fermion actions:
  - Wilson, Staggered, Domain wall, Overlap, ...



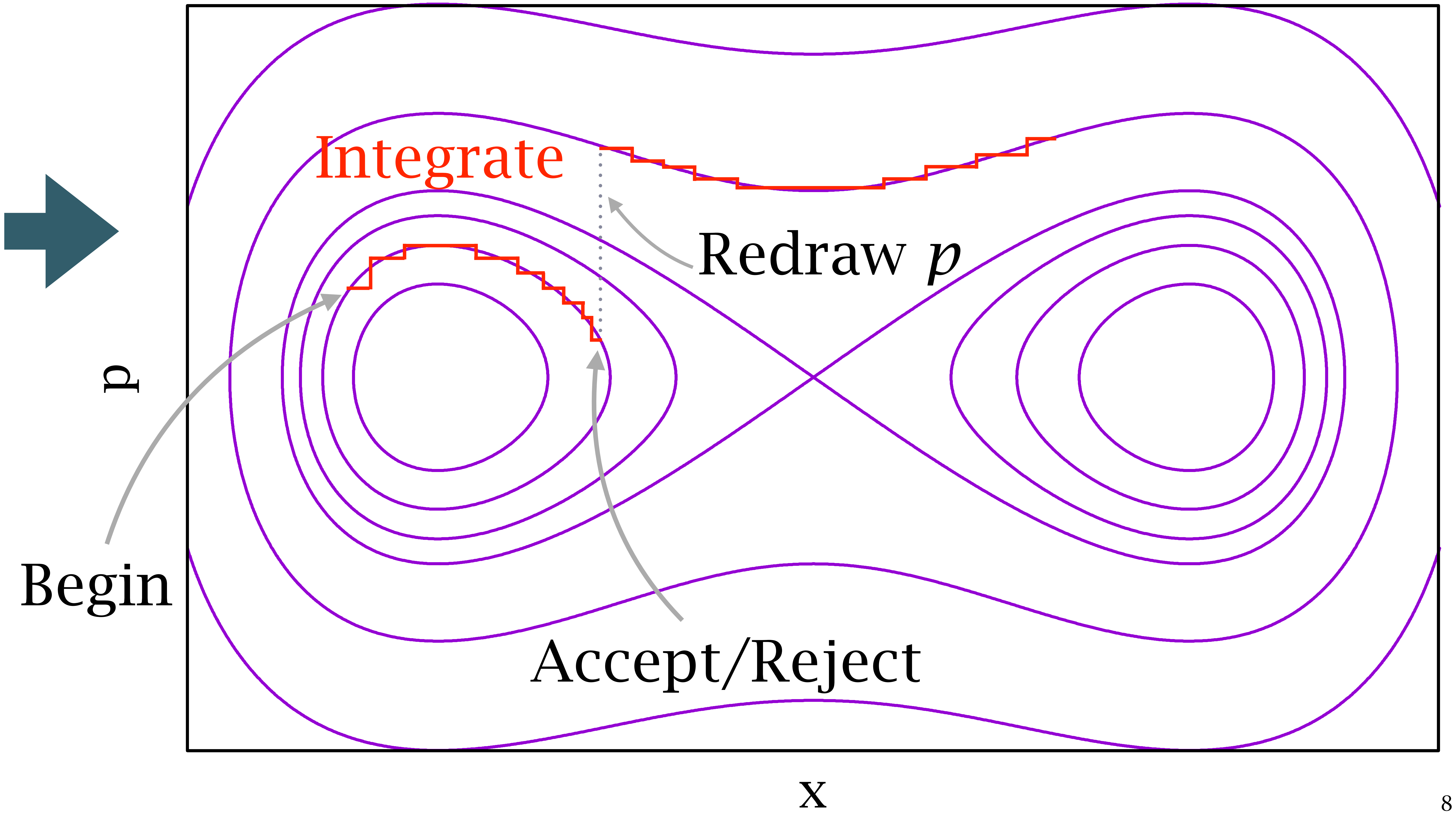
HMC

Hamiltonian dynamics moves long distances
Slow modes exist in multi-D potential
Difficult to go over potential barrier





Combined Hamiltonian (contour)



Incomplete list of HMC tricks in production

- Task: sampling space-time lattice gauge field with about tens of billions of d.o.f.
- Omelyan, higher order integrators, with force-gradient terms
- Splitting fermion determinant in multiple terms, from ratios of determinants of unequal fermion masses
  - Force from individual terms smaller
  - Large mass terms cheaper inversion
- Sexton & Weingarten, 1992, MD integration scheme
  - Larger time steps for expensive, small force terms
  - Smaller time steps for cheap, large force terms

Path integral, change of variables

- Change of variables: use a continuously differentiable bijective map $\mathcal{F}^{-1}$ from target field U to the mapped field $V = \mathcal{F}^{-1}(U)$, same group manifold for us

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int \mathcal{D}U \mathcal{O}(U) e^{-S(U)} = \frac{1}{Z} \int \mathcal{D}V \mathcal{O}(\mathcal{F}(V)) e^{-S(\mathcal{F}(V)) + \ln|\mathcal{F}_*|} \text{ where } \mathcal{F}_* = \frac{\partial \mathcal{F}(V)}{\partial V}$$

- Sample V with HMC according to the new action: Field Transformation HMC (FTHMC) $S_{\mathrm{FT}}(V) = S(\mathcal{F}(V)) - \ln|\mathcal{F}_*(V)|$

- Want the effective action to have lower potential barriers, or more uniform dynamics (smaller difference between slow and fast modes)

- The Jacobian determinant and its derivative must remain simple

Parametrized bijection map

- Gauge covariant, dynamics remain the same with local gauge transformations, $\Omega_x \in \mathrm{SU}(3)$

$$U_{x,\mu} \longrightarrow U'_{x,\mu} = \Omega_x^\dagger U_{x,\mu} \Omega_{x+\hat{\mu}}$$

- Lie group element, exponential map from the group algebra (differentials in tangent directions)

$$U_{x,\mu} \to U'_{x,\mu} = e^{\Pi_{x,\mu}} U_{x,\mu} \text{ where } \Pi_{x,\mu} = \sum_l \epsilon_l \partial_{x,\mu} W_l$$

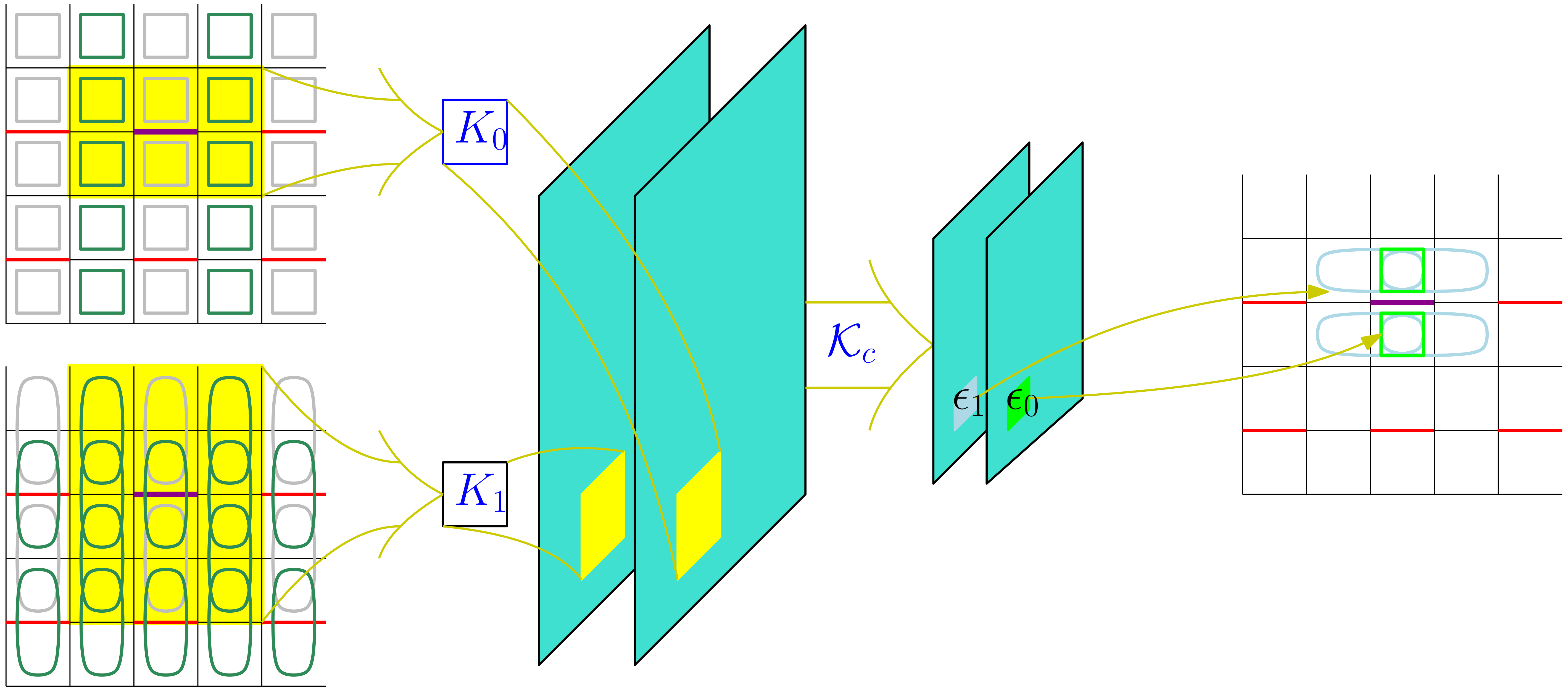
- Generalize it for machine learning
  - Make the coefficients arbitrary functions of gauge invariant quantities

$$\epsilon_{x,\mu,l} = c \tan^{-1}\left[\mathcal{N}_l(X, Y, \ldots)\right]$$

    - $X, Y, \ldots$ a list of traced Wilson loops local to x, μ, and independent of $U_{x,\mu}$
    - $\mathcal{N}$ is a convolutional neural network, $\mathcal{N}_l$ is one of the output channels
    - $c \tan^{-1}[\,\cdot\,]$ ensures a positive definite Jacobian

Localized Coefficients, by Convolutional Neural Networks

2D lattice



- Pick a subset of gauge links to update at a time (red links)
- Compute Wilson loops independent of the to-be-updated links (green loops)
- Pass through a series of convolutional neural networks and obtain coefficients

What to optimize

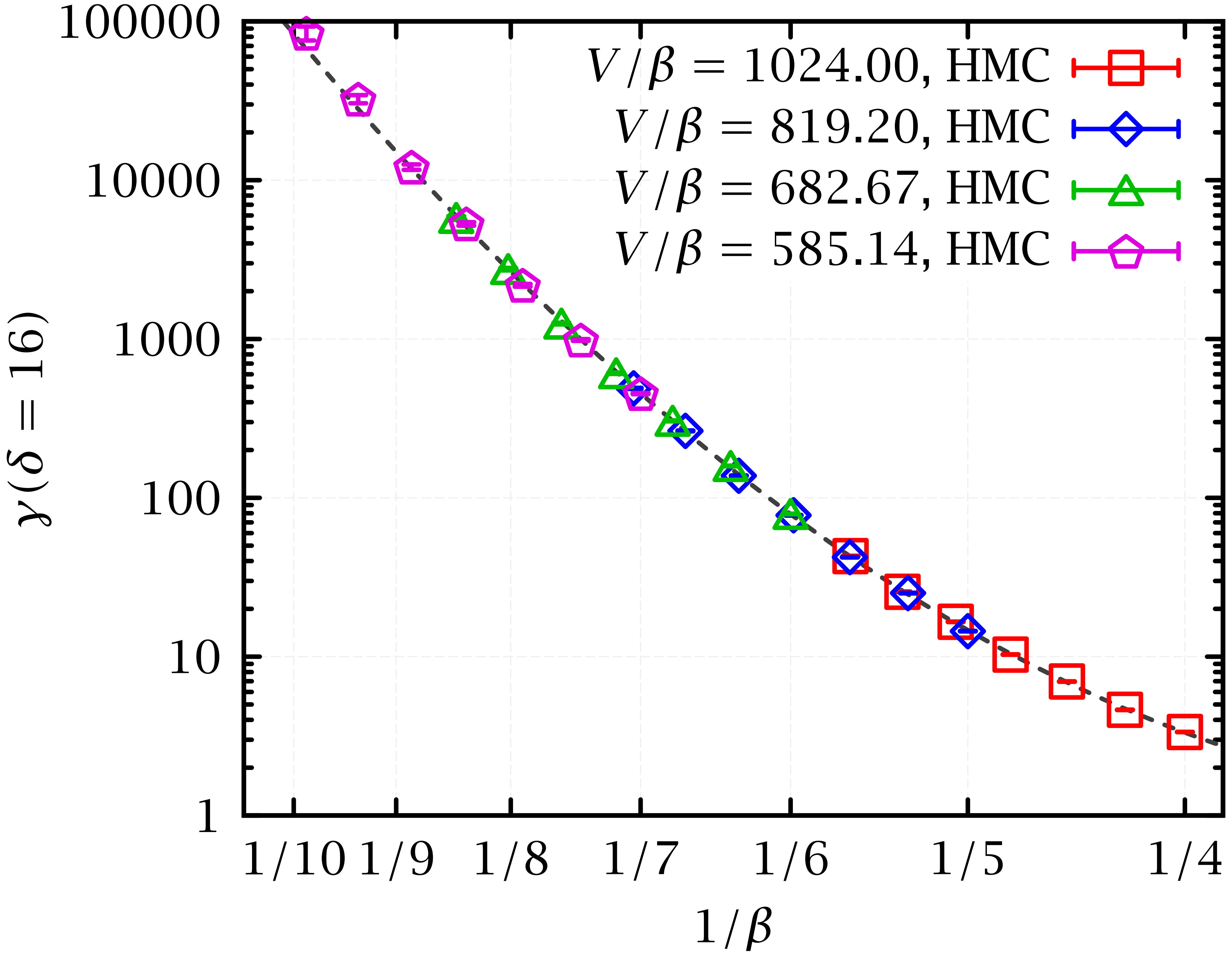
- Match the field transformed force (gradient of the effective action) against the force of the original action at a stronger coupling (away from the continuum limit)

$$\mathcal{L}(\beta, \tilde{U}) = \sum_{p \in \{2,4,6,8,10,\infty\}} \frac{c_p}{V^{1/p}} \left\| \frac{\partial S_{\mathrm{FT}}(\beta, \tilde{U})}{\partial \tilde{U}} - \frac{\partial S(\beta = 2.5, \tilde{U})}{\partial \tilde{U}} \right\|_p$$

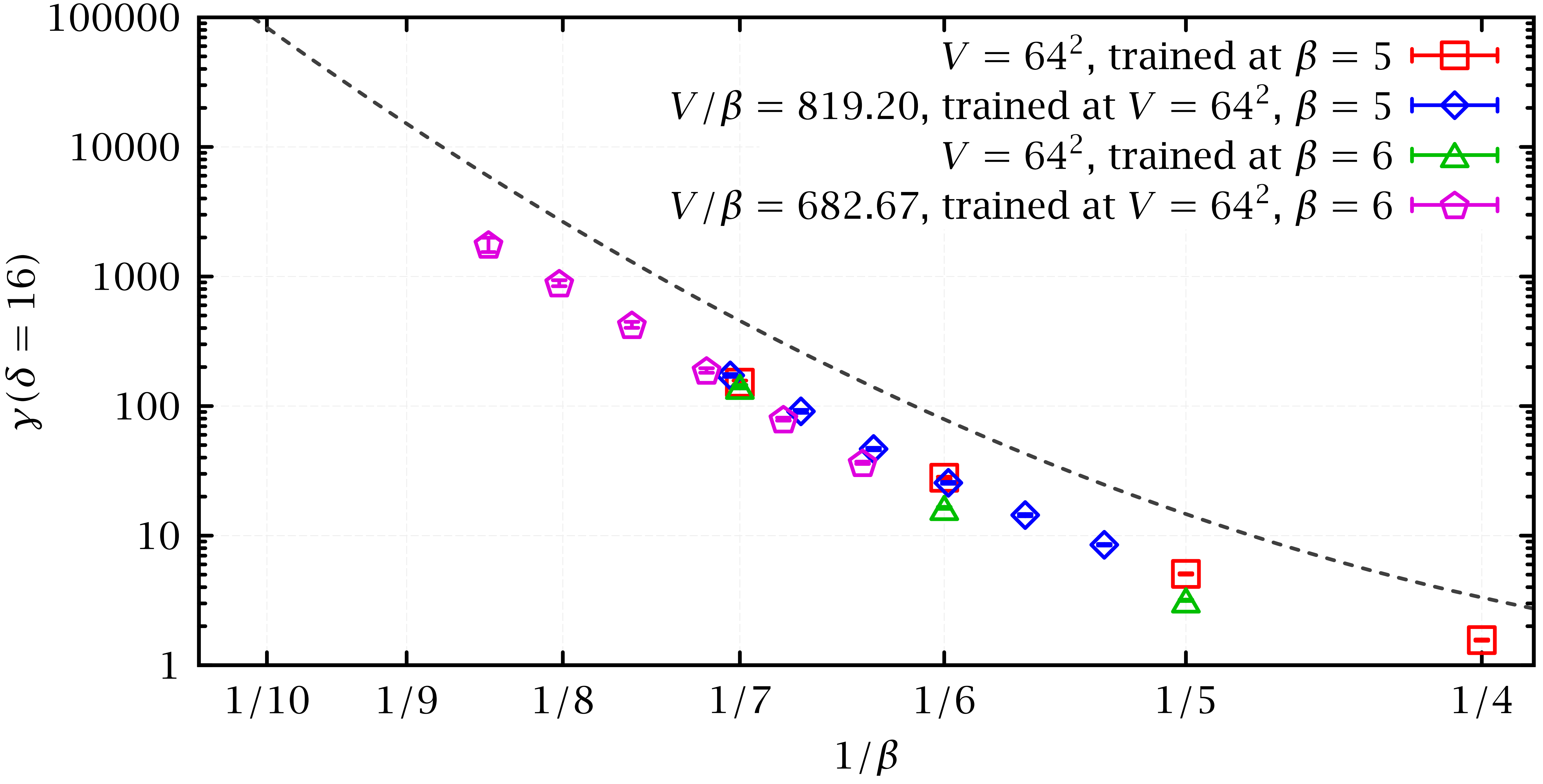
- Other possibilities under investigation
  - directly minimize the force
  - log mean exp difference
  - error term after leapfrog integration

Results from 2D U(1) lattice fields, correlation of topological charge

Scaling of the integrated autocorrelation length



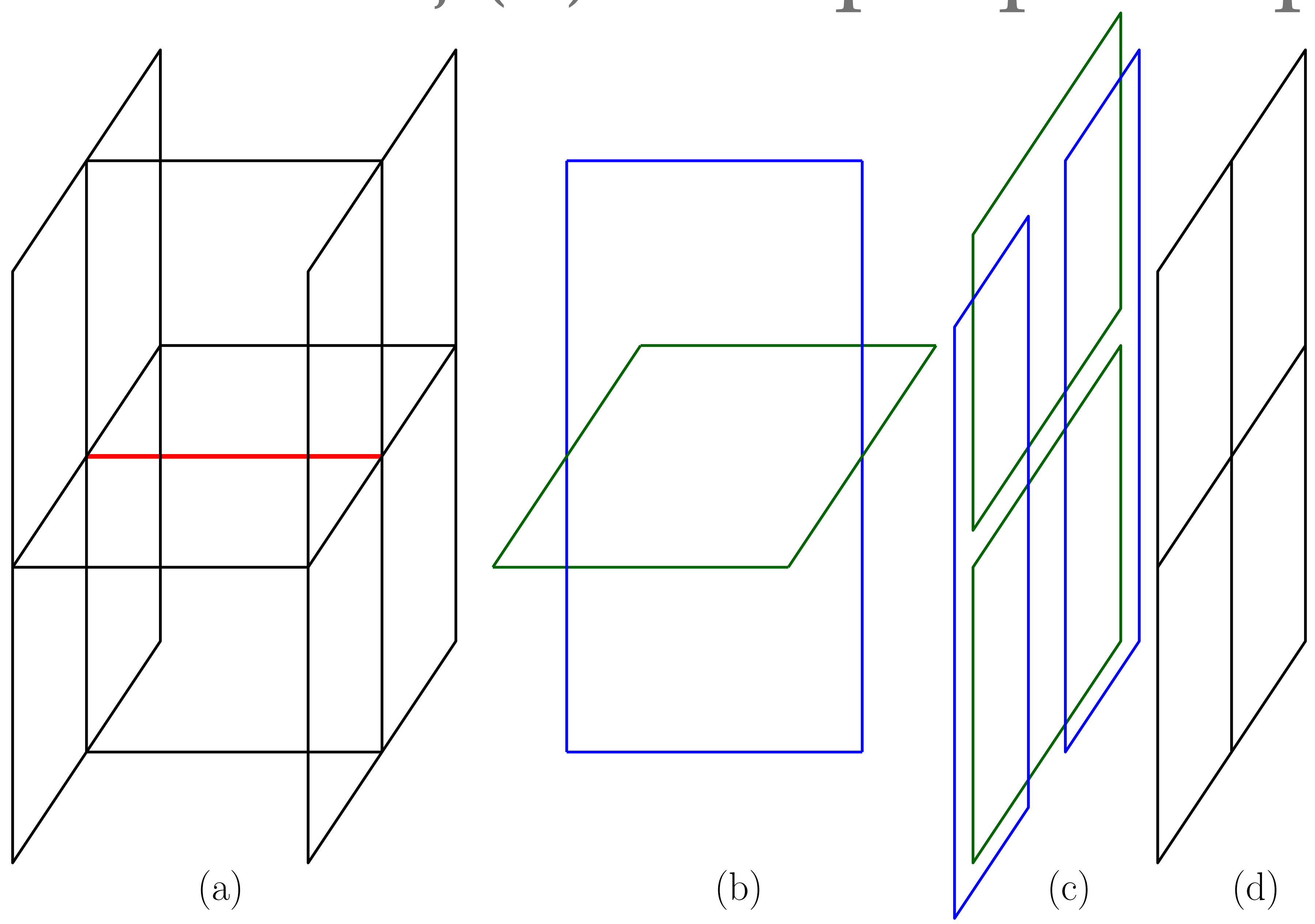
HMC



Fixed neural network architecture
Trained weights at four different conditions

Current work, on 4D SU(3) gauge fields

- The number of terms grows, and cost grows combinatorially
- for updating the red link in a 3D lattice. From left to right, (a) the links in black used to compute Wilson loops as input to a neural network, (b) two 6-link rectangle loops parallel to the red link, (c) four 6-link rectangle loops perpendicular to the red link on one side, (d) four plaquette perpendicular to the red link on one side.



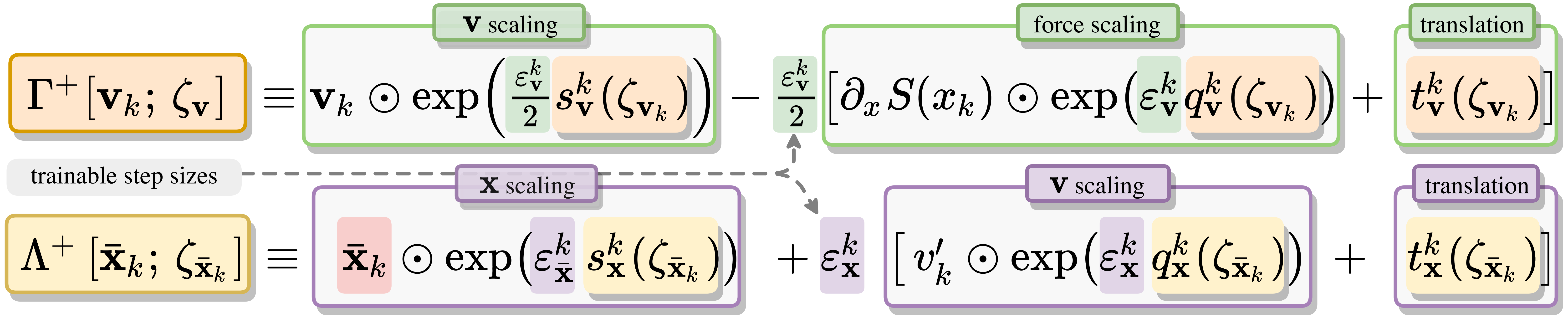
L2HMC, generalized leapfrog layers

- Original L2HMC, Levy et. al, 2017 (arXiv:1711.09268)
- We generalized it to independent leapfrog layers and adapt to gauge field

$$v' = v - \frac{\epsilon}{2}\partial_x S(x)$$

$$x' = x + \epsilon v'$$

$$v'' = v - \frac{\epsilon}{2}\partial_x S(x')$$



(c) Detailed view of the update functions Γ^+, Λ^+ for the k^{th} leapfrog step.

- Maximize the effective change of topology, $A(x^*, v^* \,|\, x, v)(Q^* - Q)^2$

Results from 2D U(1) lattices

[arXiv:2105.03418, arXiv:2112.01582]

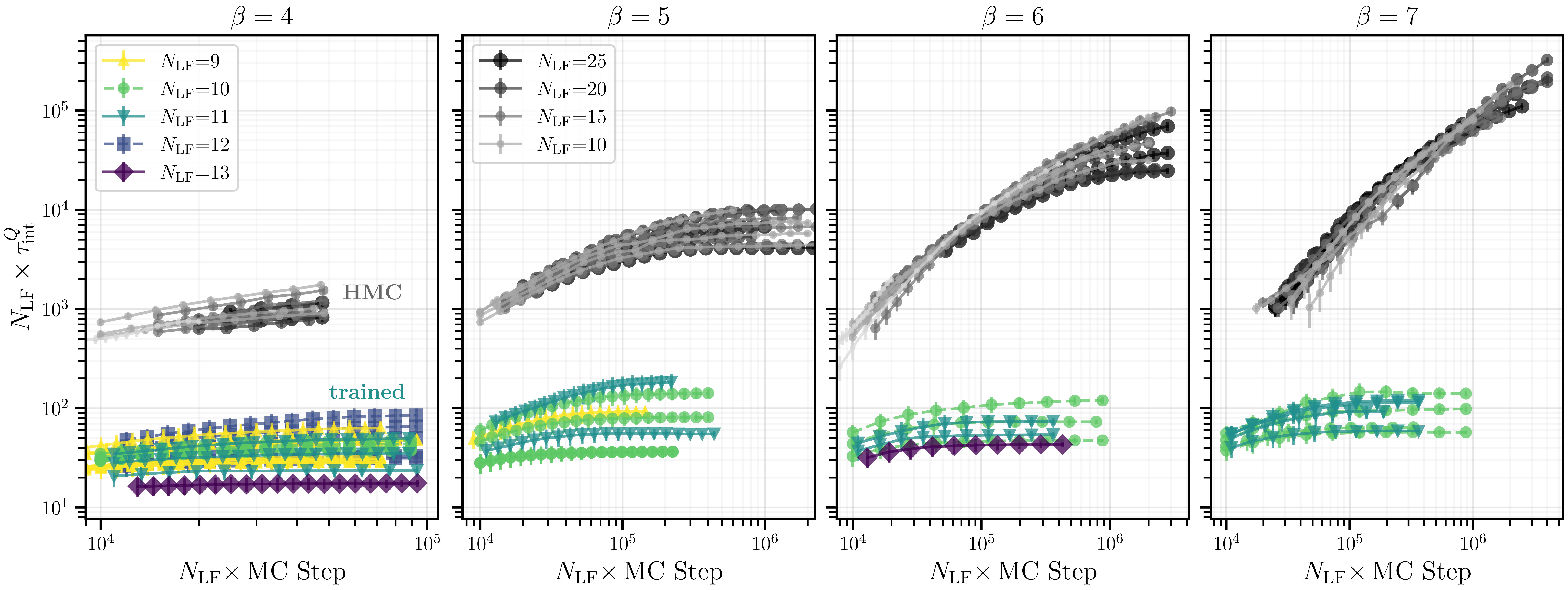
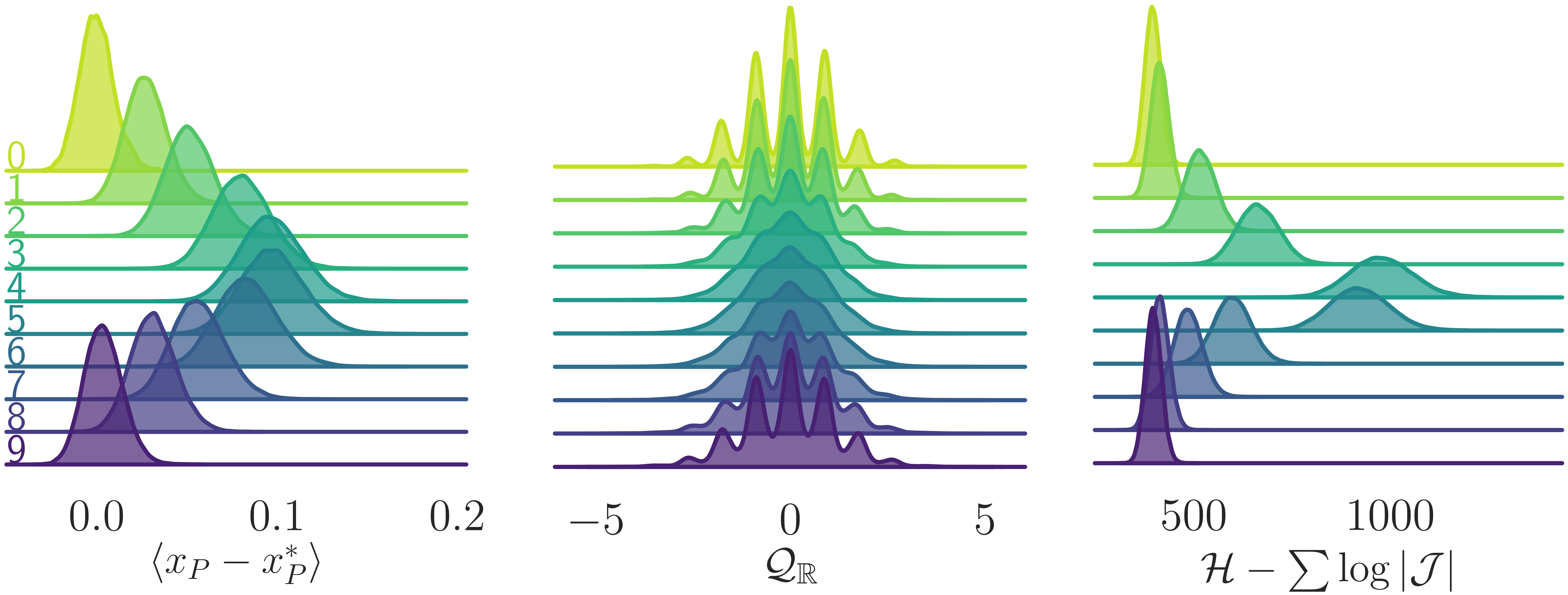


Figure 5: Comparison of the integrated autocorrelation time for trained models vs HMC with different trajectory lengths, $N_{\rm LF}$, at $\beta = 4, 5, 6, 7$ (left to right).

Dynamics per leapfrog layers



(a) Deviation in x_P. **(b)** Topological charge mixing $Q_{\mathbb{R}}$. **(c)** Artificial influx of energy.

Figure 6: Evolution of different quantities over a single trajectory consisting of $N_{\mathrm{LF}} = 10$ leapfrog steps.

Monte Carlo in propagator measurement

- Typical quark propagator, the solution of $DS(x, y) = \delta$
- Construct propagators, such as "Pion" $\langle P^a(x)P^b(y)\rangle \propto -\delta^{ab}\langle \mathrm{tr}\{S(x,y)S(x,y)^\dagger\}\rangle$
- Average as many lattice points as possible, but inversion is costly
- Use random source, with statistical error $\sim 1/\sqrt{N}$
- Class of variance reduction techniques $\langle O\rangle = \langle O - \hat{O}\rangle + \langle \hat{O}\rangle$ such that
  - $\langle O - \hat{O}\rangle$ has smaller variance; $\langle \hat{O}\rangle$ is cheaper to evaluate
  - Applies with or without random source
  - Examples: compute $\hat{O}$ using low modes of D, with approximate high modes

Outlook

- FTHMC with neural networks helps reduce autocorrelation in HMC
- L2HMC learns to inject energy in order to tunnel barriers
- Application in production requires balancing the cost and the benefit
- ML Code: https://github.com/nftqcd
- Difficulties in scaling up
  - Requires computing the Jacabian determinant and its derivatives wrt the lattice fields
  - Cost grows with spacetime dimension and the total field degrees of freedom
  - Available neural network frameworks are unprepared for 4D grid of Lie group elements; auto-grad wastes huge amount memory (size and bandwidth) in copying tensors; no optimized routines for periodic boundary conditions