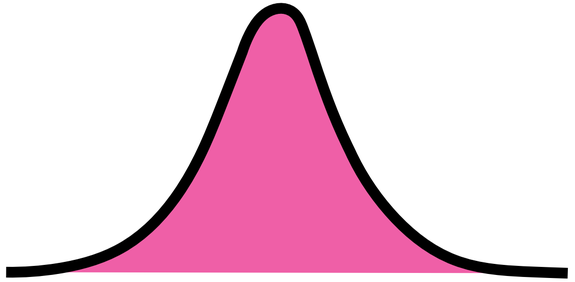
Learning the generating functional

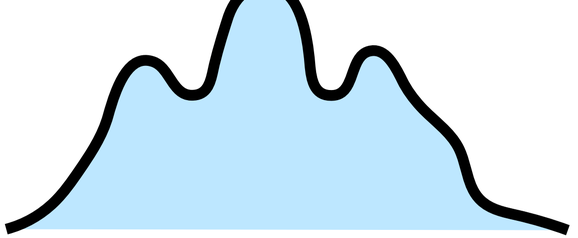
Fernando Romero-López

fernando.romero-lopez@unibe.ch

ECT, Trento*
September 9th



UNIVERSITÄT BERN



Acknowledgements

Massachusetts Institute of Technology

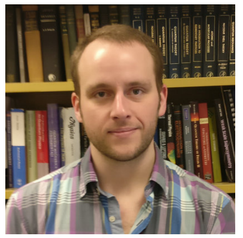

• Phiala Shanahan


• Yang Fu


• Julian Urban


• Denis Boyda

Fermilab


• Dan Hackett

[See plenary by Dan Hackett!]

u b

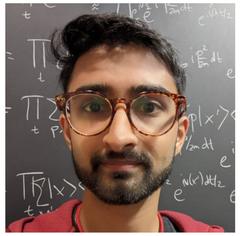
b UNIVERSITÄT BERN


• Fernando Romero-Lopez

COLUMBIA UNIVERSITY


• Ryan Abbot

THE UNIVERSITY OF EDINBURGH


• Gurtej Kanwar

Learning the generating functional for variance reduction in lattice QCD

Ryan Abbott,[1] Yang Fu,[2,3] Daniel C. Hackett,[4] Gurtej Kanwar,[5]
Fernando Romero-López,[6] and Phiala E. Shanahan[2,3]

arXiv:2606.15986

Platform for Advanced Scientific Computing

Schweizerischer Nationalfonds

Flows in Lattice QCD

Can we incorporate machine-learned flows in the lattice QCD pipeline to accelerate calculations?

Flows in Lattice QCD

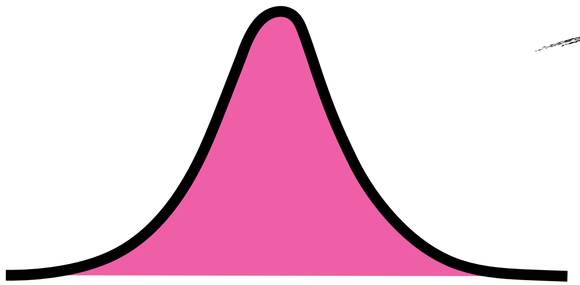
Can we incorporate machine-learned flows in the lattice QCD pipeline to accelerate calculations?

- **Two paradigms:** [Related talks by G. Kanwar, M. Gerdes, B. Lucini, S. Nakajima, A. Singha, L. Funcke, and S. Singh]
  - **Accelerate sampling, i.e. reduce cost to obtain independent MCMC samples**
    - Direct sampling via normalizing flows [Albergo, Kanwar, Shanahan 1904.12072]
    - Complementary approaches like Parallel Tempering or out-of-equilibrium methods [Abbott et al, 2404.11674] [Bonanno et al, 2510.25704] [Talk by L. Verzichelli]
    - Multi-scale methods [Hasenfratz et al, 2608.28581] [Talk by E. Cellini, A. Hasenfratz]
  - **Reduce measurement costs, i.e. construct improved estimators for a fixed existing ensemble**
    - Variance reduction via normalizing flows [This talk!] [See related talk by N. Hermansson-Truedsson] [See related talk by P. Butti @ LAT26]

Flow models

[Rezende, Mohamed, 1505.05770]

$r(z)$

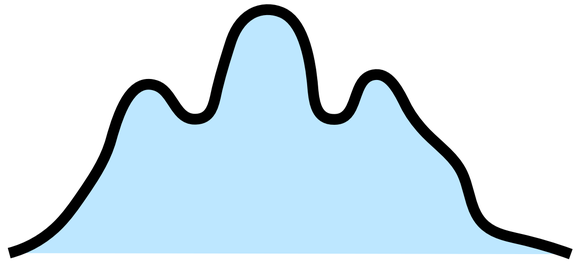


easier-to-sample
distribution
(prior)

$f(z)$

tractable Jacobian
&
invertible

$q(\phi) \simeq p(\phi)$



approximates
target distribution
(model)

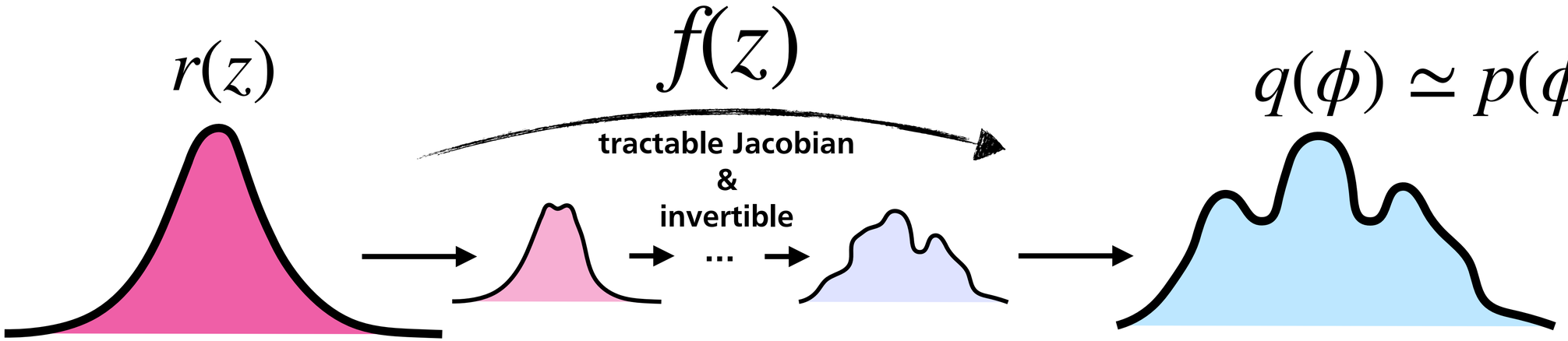
"Trainable change of variables"

Model probability

$$q(\phi) = r(z)\left|\det \frac{\partial f(z)}{\partial z}\right|^{-1}$$

Flow models

[Rezende, Mohamed, 1505.05770]



easier-to-sample distribution (prior)

approximates target distribution (model)

parametrized by neural networks (trainable and expressive)

composed of many simple layers

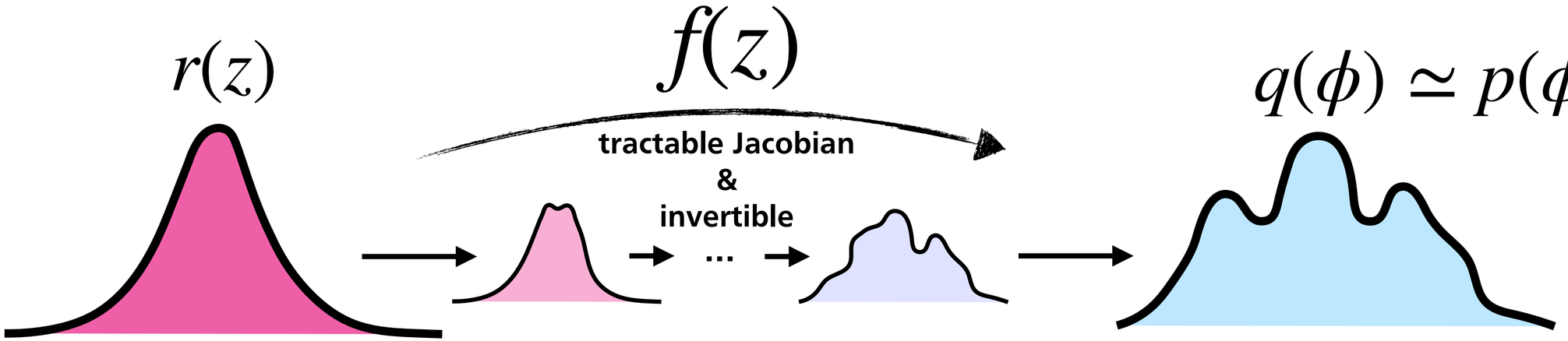
"Trainable change of variables"

Model probability

$$q(\phi) = r(z)\left|\det \frac{\partial f(z)}{\partial z}\right|^{-1}$$

Flow models

[Rezende, Mohamed, 1505.05770]



easier-to-sample distribution (prior)

approximates target distribution (model)

parametrized by neural networks (trainable and expressive)

composed of many simple layers

"Trainable change of variables"

Model probability

$$q(\phi) = r(z)\left|\det \frac{\partial f(z)}{\partial z}\right|^{-1}$$

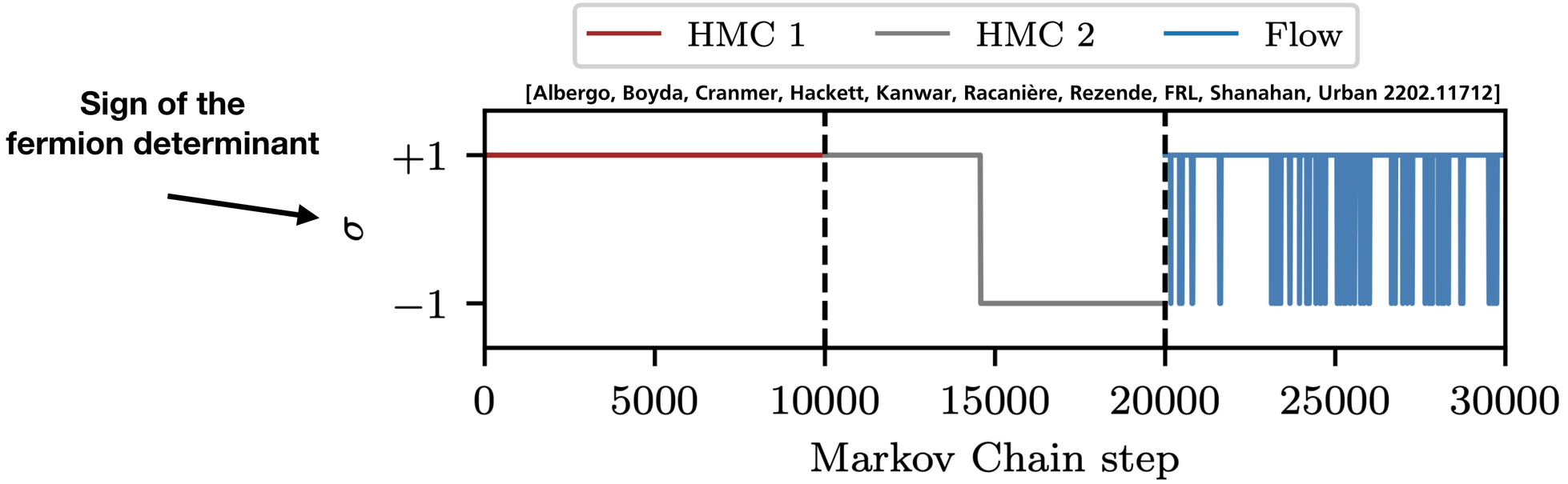
❗ Trained models are not perfect.

☑ Exact sampling can be recovered via Markov Chain/reweighting

Flows for (direct) sampling

- **Original LQCD applications of flows aimed to tackle Critical Slowing Down/Topology Freezing**
  [Albergo, Kanwar, Shanahan 1904.12072]

- **Promising results in low-dimensional theories (Schwinger model)**



Our recent works

(1+1)d real scalar field theory
[Albergo, Kanwar, Shanahan 1904.12072]
[Hackett, Hsieh, Albergo, Boyda, JW Chen, KF Chen, Cranmer, Kanwar, Shananan 2107.00734]

(1+1)d Abelian gauge theory
[Kanwar, Albergo, Boyda, Cranmer, Hackett, Racanière, Rezende, Shanahan 2003.06413]

(1+1)d non-Abelian gauge theory
[Kanwar, Albergo, Boyda, Cranmer, Hackett, Racanière, Rezende, Shanahan 2003.06413]

(1+1)d Yukawa model
i.e. real scalar field theory + fermions
[Albergo, Kanwar, Racanière, Rezende, Urban, Boyda, Cranmer, Hackett, Shanahan 2106.05934]

Schwinger model i.e. (1+1)d QED
[Albergo, Boyda, Cranmer, Hackett, Kanwar, Racanière, Rezende, FRL, Shanahan, Urban 2202.11712]

2D fermionic gauge theories with pseudofermions
[Abbott, Albergo, Boyda, Cranmer, Hackett, Kanwar, Racanière, Rezende, FRL, Shanahan, Tian, Urban 2207.08945]

QCD/SU(3) in the strong-coupling region
[Abbott et al, 2208.03832] [Abbott et al, 2305.02402]

...

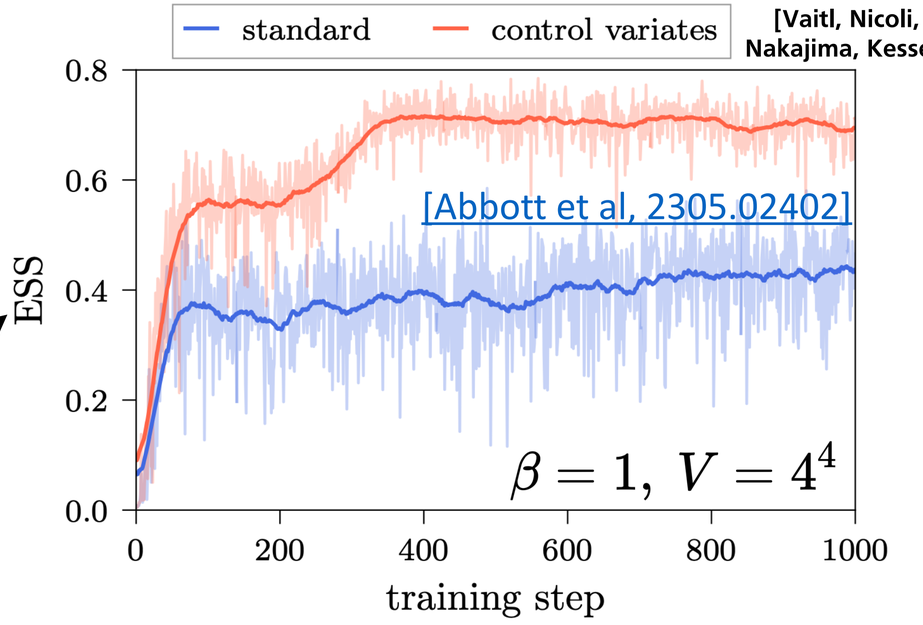
Developments are still needed for large-scale QCD

Flow models

- Suitable architectures for 4D gauge theories.
  - However, direct sampling remains hard
  - Need very high-quality models to reach large volumes
    (Naive volume scaling is exponential)

model quality metric

$$\mathrm{ESS} = \frac{\langle w \rangle^2}{\langle w^2 \rangle}$$

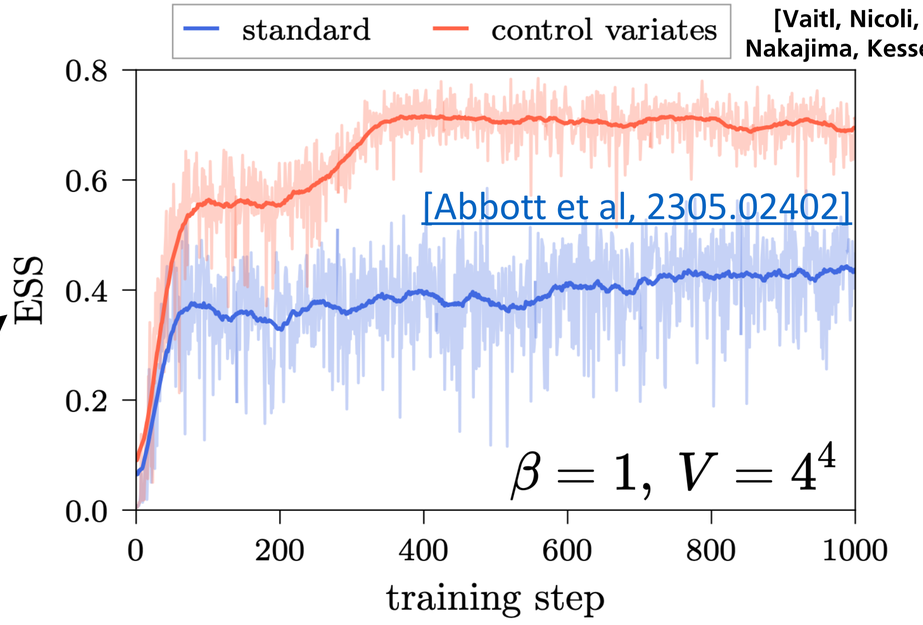


Flow models

- Suitable architectures for 4D gauge theories.
  - However, direct sampling remains hard
  - Need very high-quality models to reach large volumes
    (Naive volume scaling is exponential)

model quality metric

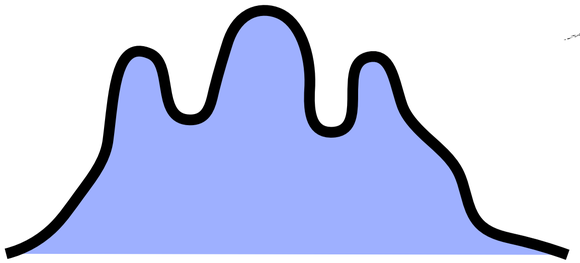
$$\mathrm{ESS} = \frac{\langle w \rangle^2}{\langle w^2 \rangle}$$



- Instead, explore applications with smaller gap between theories: $\mathrm{QCD} \to \mathrm{QCD} + \lambda \mathcal{O}$

Short-distance flows

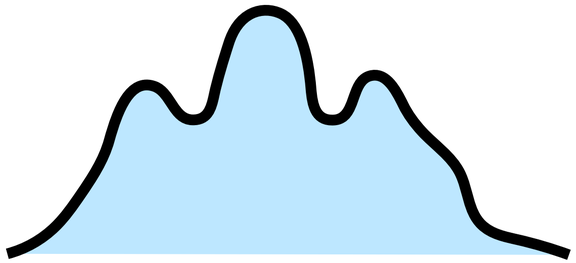
$p_0(U)$



$$U' = f(U)$$

tractable Jacobian
&
invertible

$q(U') \simeq p_\lambda(U')$



Non-trivial prior
sampled via MCMC

model approximates
target distribution

☑ Target is close to prior → Current architectures can already make a difference !

Flows for variance reduction

☑ **A promising application for flows is variance reduction of lattice QCD observables**

Applications of flow models to the generation of correlated lattice QCD ensembles

Ryan Abbott,[1,2] Aleksandar Botev,[3] Denis Boyda,[1,2] Daniel C. Hackett,[4,1,2] Gurtej Kanwar,[5] Sébastien Racanière,[3] Danilo J. Rezende,[3] Fernando Romero-López,[1,2] Phiala E. Shanahan,[1,2] and Julian M. Urban[1,2]

arXiv:2401.10874

Variance reduction in lattice QCD observables via normalizing flows

Ryan Abbott,[1,2,3] Denis Boyda,[2,3] Yang Fu,[2,3] Daniel C. Hackett,[4] Gurtej Kanwar,[5] Fernando Romero-López,[6] Phiala E. Shanahan,[2,3] and Julian M. Urban[2,3]

arXiv:2603.02984

Learning the generating functional for variance reduction in lattice QCD

Ryan Abbott,[1] Yang Fu,[2,3] Daniel C. Hackett,[4] Gurtej Kanwar,[5] Fernando Romero-López,[6] and Phiala E. Shanahan[2,3]

arXiv:2606.15986

Flows for variance reduction

- A promising application for flows is variance reduction of lattice QCD observables

Applications of flow models to the generation of correlated lattice QCD ensembles

Ryan Abbott,[1,2] Aleksandar Botev,[3] Denis Boyda,[1,2] Daniel C. Hackett,[4,1,2] Gurtej Kanwar,[5] Sébastien Racanière,[3] Danilo J. Rezende,[3] Fernando Romero-López,[1,2] Phiala E. Shanahan,[1,2] and Julian M. Urban[1,2]

arXiv:2401.10874

Variance reduction in lattice QCD observables via normalizing flows

Ryan Abbott,[1,2,3] Denis Boyda,[2,3] Yang Fu,[2,3] Daniel C. Hackett,[4] Gurtej Kanwar,[5] Fernando Romero-López,[6] Phiala E. Shanahan,[2,3] and Julian M. Urban[2,3]

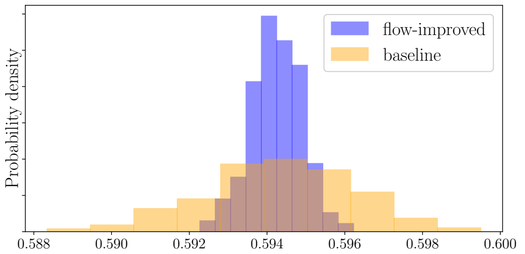
arXiv:2603.02984

Learning the generating functional for variance reduction in lattice QCD

Ryan Abbott,[1] Yang Fu,[2,3] Daniel C. Hackett,[4] Gurtej Kanwar,[5] Fernando Romero-López,[6] and Phiala E. Shanahan[2,3]

arXiv:2606.15986

- Main goal: compute unbiased estimators with lower statistical error at fixed number of configs



Improved observables

- Variance reduction for observables that can be formulated with derivatives with respect to action parameters

$$\frac{d\langle \mathcal{O} \rangle}{d\alpha}$$

action parameter

  - For instance, matrix element using Feynman-Hellmann techniques: sigma terms, hadron structure.

Improved observables

- Variance reduction for observables that can be formulated with derivatives with respect to action parameters

$$\frac{d\langle \mathcal{O} \rangle}{d\alpha}$$

action parameter

  - For instance, matrix element using Feynman-Hellmann techniques: sigma terms, hadron structure.

More generally: any N-point function via generating functional

Inspiration: [Catumba, Ramos 2502.15570]

Improved observables

- Variance reduction for observables that can be formulated with derivatives with respect to action parameters

$$\frac{d\langle \mathcal{O} \rangle}{d\alpha}$$

action parameter

  - For instance, matrix element using Feynman-Hellmann techniques: sigma terms, hadron structure.

More generally: any N-point function via generating functional

Inspiration: [Catumba, Ramos 2502.15570]

- Interpretation as "control variate": same expectation value but difference variance

$$\langle \mathcal{Q} \rangle \to \langle \mathcal{Q} \rangle + \langle \nabla \cdot F - F \cdot \nabla S_0 \rangle \equiv \langle \mathcal{Q} \rangle + \langle \mathcal{L}(F) \rangle$$

$$\langle \mathcal{L}(F) \rangle = 0, \quad \mathrm{Cov}(\mathcal{L}(F), \mathcal{Q}) \neq 0$$

Outline

1. The generating functional in LFT
2. Learning the generating functional
3. Residual architectures for gauge theories
4. Numerical application: Wilson loops & glueballs
5. Application to meson correlators: σ meson

The generating functional in Lattice Field Theory

Generating functional

- Variance reduction can be framed by means of the generating functional

expectation value in λ=0 theory

$$Z_{\lambda_1,\ldots,\lambda_n} = \int DU e^{-S_{\mathrm{QCD}}(U)+\sum_i \lambda_i \mathcal{O}_i} \xrightarrow{\text{Ratio of partition functions}} \frac{Z_{\lambda_1,\ldots,\lambda_n}}{Z_0} = \left\langle e^{\sum_i \lambda_i \mathcal{O}_i} \right\rangle$$

Generating functional

- Variance reduction can be framed by means of the generating functional

expectation value in λ=0 theory

$$Z_{\lambda_1,\ldots,\lambda_n} = \int DU e^{-S_{\mathrm{QCD}}(U)+\sum_i \lambda_i \mathcal{O}_i} \xrightarrow{\text{Ratio of partition functions}} \frac{Z_{\lambda_1,\ldots,\lambda_n}}{Z_0} = \left\langle e^{\sum_i \lambda_i \mathcal{O}_i} \right\rangle$$

- ☑ N-point function can be generated via derivatives with respect to small parameters

$$\langle \mathcal{O}_1 \ldots \mathcal{O}_n \rangle = \frac{\partial^n}{\partial\lambda_1 \ldots \partial\lambda_n} \frac{Z_{\lambda_1,\ldots,\lambda_n}}{Z_0} \bigg|_{\lambda_i=0}$$

Generating functional

- Variance reduction can be framed by means of the generating functional

$$Z_{\lambda_1,\ldots,\lambda_n} = \int DU e^{-S_{\mathrm{QCD}}(U)+\sum_i \lambda_i \mathcal{O}_i} \xrightarrow{\text{Ratio of partition functions}} \frac{Z_{\lambda_1,\ldots,\lambda_n}}{Z_0} = \left\langle e^{\sum_i \lambda_i \mathcal{O}_i} \right\rangle$$

expectation value in λ=0 theory

- ☑ N-point function can be generated via derivatives with respect to small parameters

$$\langle \mathcal{O}_1 \ldots \mathcal{O}_n \rangle = \frac{\partial^n}{\partial\lambda_1 \ldots \partial\lambda_n} \frac{Z_{\lambda_1,\ldots,\lambda_n}}{Z_0} \bigg|_{\lambda_i=0}$$

- Computing ratios of partition functions can be seen as a reweighting problem

$$\frac{Z_{\lambda_1,\ldots,\lambda_n}}{Z_0} = \left\langle e^{\sum_i \lambda_i \mathcal{O}_i} \right\rangle \equiv \left\langle \hat{w}_{\lambda_1,\ldots,\lambda_n} \right\rangle$$

unnormalized reweighting factor

Example: one-point functions

- Example: one-point function

$$\langle \hat{w}_\lambda \rangle = \frac{Z_\lambda}{Z_0}$$

$$\langle \mathcal{O} \rangle = \partial_\lambda \langle \hat{w}_\lambda \rangle = \partial_\lambda \langle e^{\lambda \mathcal{O}} \rangle = \lim_{\lambda \to 0} \frac{\langle \hat{w}_\lambda \rangle - 1}{\lambda}$$

Example: one-point functions

- Example: one-point function

$$\langle \hat{w}_\lambda \rangle = \frac{Z_\lambda}{Z_0}$$

$$\langle \mathcal{O} \rangle = \partial_\lambda \langle \hat{w}_\lambda \rangle = \partial_\lambda \langle e^{\lambda \mathcal{O}} \rangle = \lim_{\lambda \to 0} \frac{\langle \hat{w}_\lambda \rangle - 1}{\lambda}$$

- Introduce "Effective Sample Size per configuration" metric:

$$\mathrm{ESS} = \frac{\langle w_\lambda \rangle^2}{\langle w_\lambda^2 \rangle}$$

Quantifies fluctuations in reweighting factors

$$\mathrm{ESS} \in (0, 1]$$

- The variance of a one-point function is:

$$\mathrm{Var}[\mathcal{O}] = \mathrm{Var}\left[\partial_\lambda \hat{w}_\lambda\right] = \lim_{\lambda \to 0} \frac{1}{\lambda^2} \left(\frac{1}{\mathrm{ESS}} - 1 \right) = \mathrm{const} + O(\lambda)$$

Example: one-point functions

- Example: one-point function

$$\langle \hat{w}_\lambda \rangle = \frac{Z_\lambda}{Z_0}$$

$$\langle \mathcal{O} \rangle = \partial_\lambda \langle \hat{w}_\lambda \rangle = \partial_\lambda \langle e^{\lambda \mathcal{O}} \rangle = \lim_{\lambda \to 0} \frac{\langle \hat{w}_\lambda \rangle - 1}{\lambda}$$

- Introduce "Effective Sample Size per configuration" metric:

$$\mathrm{ESS} = \frac{\langle w_\lambda \rangle^2}{\langle w_\lambda^2 \rangle}$$

Quantifies fluctuations in reweighting factors

$$\mathrm{ESS} \in (0, 1]$$

- The variance of a one-point function is:

$$\mathrm{Var}[\mathcal{O}] = \mathrm{Var}\left[\partial_\lambda \hat{w}_\lambda\right] = \lim_{\lambda \to 0} \frac{1}{\lambda^2} \left(\frac{1}{\mathrm{ESS}} - 1 \right) = \mathrm{const} + O(\lambda)$$

Natural transition into the flow approach: increasing the ESS leads to lower variance!

N-point functions

- **Method 1: "derivative trick" to obtain (N+1)-point function from N-point function**
  [Catumba, Ramos 2502.15570]

$$\langle \mathcal{O}\mathcal{J}\rangle - \langle \mathcal{O}\rangle\langle \mathcal{J}\rangle = \frac{d\langle \mathcal{O}\rangle_\lambda}{d\lambda}\bigg|_{\lambda=0} = \langle \mathcal{O}\,\partial_\lambda w_\lambda\rangle|_{\lambda=0} \simeq \frac{\langle \mathcal{O}w_\lambda\rangle - \langle \mathcal{O}\rangle}{\lambda}$$

Normalized reweighting factor

$$w_\lambda = \frac{\hat{w}_\lambda}{\langle \hat{w}_\lambda\rangle}$$

N-point functions

- **Method 1: "derivative trick" to obtain (N+1)-point function from N-point function**
  [Catumba, Ramos 2502.15570]

$$\langle \mathcal{O}\mathcal{J}\rangle - \langle \mathcal{O}\rangle\langle \mathcal{J}\rangle = \left.\frac{d\langle \mathcal{O}\rangle_\lambda}{d\lambda}\right|_{\lambda=0} = \langle \mathcal{O}\,\partial_\lambda w_\lambda\rangle|_{\lambda=0} \simeq \frac{\langle \mathcal{O}w_\lambda\rangle - \langle \mathcal{O}\rangle}{\lambda}$$

Normalized reweighting factor

$$w_\lambda = \frac{\hat{w}_\lambda}{\langle \hat{w}_\lambda\rangle}$$

- **Method 2: "generating functional" approach to generate N-point functions from zero-point functions**

$$\langle \mathcal{O}_k\mathcal{O}_\ell\rangle = \frac{d^2\left\langle \hat{w}_{\lambda_k,\lambda_\ell}\right\rangle}{d\lambda_k d\lambda_\ell} \qquad \text{where} \qquad \hat{w}_{\lambda_k,\lambda_\ell} = e^{\lambda_k\mathcal{O}_k+\lambda_\ell\mathcal{O}_\ell}$$

N-point functions

- **Method 1: "derivative trick" to obtain (N+1)-point function from N-point function**
  [Catumba, Ramos 2502.15570]

$$\langle \mathcal{O}\mathcal{J}\rangle - \langle \mathcal{O}\rangle\langle \mathcal{J}\rangle = \frac{d\langle \mathcal{O}\rangle_\lambda}{d\lambda}\bigg|_{\lambda=0} = \langle \mathcal{O}\,\partial_\lambda w_\lambda\rangle|_{\lambda=0} \simeq \frac{\langle \mathcal{O}w_\lambda\rangle - \langle \mathcal{O}\rangle}{\lambda}$$

Normalized reweighting factor

$$w_\lambda = \frac{\hat{w}_\lambda}{\langle \hat{w}_\lambda\rangle}$$

- **Method 2: "generating functional" approach to generate N-point functions from zero-point functions**

$$\langle \mathcal{O}_k\mathcal{O}_\ell\rangle = \frac{d^2\left\langle \hat{w}_{\lambda_k,\lambda_\ell}\right\rangle}{d\lambda_k d\lambda_\ell} \qquad \text{where} \qquad \hat{w}_{\lambda_k,\lambda_\ell} = e^{\lambda_k\mathcal{O}_k+\lambda_\ell\mathcal{O}_\ell}$$

- **Generating functional: Higher potential for variance reduction:** $\mathrm{Var}[\mathcal{O}_k\mathcal{O}_\ell] \simeq \mathrm{Var}[\mathcal{O}_k]\mathrm{Var}[\mathcal{O}_\ell] + \ldots$

(For the independent variables, e.g. at large time separations)

Learning the generating functional

Learned generating functional

- We train a flow to learn the insertion of an operator:

model prob density →

$$q(V) = r(U)|\det J_f|^{-1} \approx \frac{1}{Z_\lambda} e^{-S(V)+\lambda\mathcal{O}(V)}.$$

Learned generating functional

- We train a flow to learn the insertion of an operator:

model prob density

$$q(V) = r(U)|\det J_f|^{-1} \approx \frac{1}{Z_\lambda} e^{-S(V)+\lambda\mathcal{O}(V)}.$$

- Generating functional approach still holds

$$\langle O \rangle = \langle \partial_\lambda \hat{w}_\lambda \rangle \simeq \frac{\langle \hat{w}_\lambda \rangle - 1}{\lambda}$$

unnormalized reweighting factor on the flowed fields!

- Importantly, reweighting factors now include the Jacobian of the flow

$$\hat{w}_\lambda(U) \equiv e^{-S(V)+\lambda\mathcal{O}(V)+S(U)}|\det J_f|, \qquad V = f(U)$$

Learned generating functional

- We train a flow to learn the insertion of an operator:

model prob density →

$$q(V) = r(U)|\det J_f|^{-1} \approx \frac{1}{Z_\lambda} e^{-S(V)+\lambda\mathcal{O}(V)}.$$

- Generating functional approach still holds

$$\langle O \rangle = \langle \partial_\lambda \hat{w}_\lambda \rangle \simeq \frac{\langle \hat{w}_\lambda \rangle - 1}{\lambda}$$

← unnormalized reweighting factor on the flowed fields!

- Importantly, reweighting factors now include the Jacobian of the flow

$$\hat{w}_\lambda(U) \equiv e^{-S(V)+\lambda\mathcal{O}(V)+S(U)}|\det J_f|, \qquad V = f(U)$$

- Guarantee of exactness: flows do not introduce any bias

$$\langle \hat{w}_\lambda \rangle = \frac{Z_\lambda}{Z_0} \qquad \text{for any invertible flow}$$

Perfect flow limit

- Recall the one-point function expression:

$$\langle O \rangle = \langle \partial_\lambda \hat{w}_\lambda \rangle \simeq \frac{\langle \hat{w}_\lambda \rangle - 1}{\lambda}$$

- A perfect flow, i.e. ESS=1, leads to noiseless estimators

$$\mathrm{Var}[\mathcal{O}] = \mathrm{Var}[\partial_\lambda \hat{w}_\lambda] = \lim_{\lambda \to 0} \frac{1}{\lambda^2} \left(\frac{1}{\mathrm{ESS}} - 1 \right) \overset{\text{perfect flow}}{=} 0$$

- Key point:

$$\hat{w}_\lambda(U) \xrightarrow{\text{perfect flow}} \frac{Z_\lambda}{Z_0} \text{ configuration by configuration.}$$

Perfect flow limit

- Recall the one-point function expression: $$\langle O \rangle = \langle \partial_\lambda \hat{w}_\lambda \rangle \simeq \frac{\langle \hat{w}_\lambda \rangle - 1}{\lambda}$$

- A perfect flow, i.e. ESS=1, leads to noiseless estimators

$$\mathrm{Var}[\mathcal{O}] = \mathrm{Var}[\partial_\lambda \hat{w}_\lambda] = \lim_{\lambda \to 0} \frac{1}{\lambda^2} \left(\frac{1}{\mathrm{ESS}} - 1 \right) \overset{\text{perfect flow}}{=} 0$$

- Key point:

$$\hat{w}_\lambda(U) \xrightarrow{\text{perfect flow}} \frac{Z_\lambda}{Z_0} \text{ configuration by configuration.}$$

- Exact flow must exist as a connection between nearby distributions
  - In practice, hard to achieve but high-quality flows are possible $\mathrm{ESS}_{\mathrm{flow}} > \mathrm{ESS}_{\mathrm{identity}}$

Infinitesimal picture

- Consider a flow close to the identity with an explicit λ dependence

Flow field

$$V = f(U) = e^{\lambda F(U)} U \simeq \{1 + \lambda F(U)\} U$$

Infinitesimal picture

- Consider a flow close to the identity with an explicit λ dependence

Flow field

$$V = f(U) = e^{\lambda F(U)} U \simeq \{1 + \lambda F(U)\} U$$

- Expand the reweighting factor:

$$\hat{w}_\lambda(U) \equiv e^{\lambda \mathcal{O}(V) - [S(V) - S(U)] + \log\det J_f}$$

$$\hat{w}_\lambda(U) \simeq 1 + \lambda(\mathcal{O}(U) - F \cdot \nabla S + \nabla \cdot F) + \dots$$

Infinitesimal picture

- Consider a flow close to the identity with an explicit λ dependence

Flow field

$$V = f(U) = e^{\lambda F(U)} U \simeq \{1 + \lambda F(U)\} U$$

- Expand the reweighting factor:

$$\hat{w}_\lambda(U) \equiv e^{\lambda \mathcal{O}(V) - [S(V) - S(U)] + \log \det J_f}$$

$$\hat{w}_\lambda(U) \simeq 1 + \lambda(\mathcal{O}(U) - F \cdot \nabla S + \nabla \cdot F) + \dots$$

- Improved estimator of the one-point function in the control variate picture

$$\langle \partial_\lambda \hat{w}_\lambda \rangle = \langle \mathcal{O} + \nabla \cdot F - F \cdot \nabla S \rangle \equiv \langle \mathcal{O} \rangle + \langle \mathcal{L}(F) \rangle$$

$= 0$

Improved two-point functions

- **Two-point function as:** $C(t) = \langle \mathcal{G}(t)\mathcal{G}(0)\rangle_c = \langle \mathcal{G}(t)\mathcal{G}(0)\rangle - \langle \mathcal{G}(t)\rangle\langle \mathcal{G}(0)\rangle$

finite-difference second derivative →

$$C(t) \simeq \frac{1}{\lambda\lambda'}\left[\left\langle \hat{w}_{\lambda,\lambda'}\right\rangle - \left\langle \hat{w}_{\lambda,0}\right\rangle\left\langle \hat{w}_{0,\lambda'}\right\rangle\right]$$

$$\left\langle \hat{w}_{\lambda,\lambda'}\right\rangle = \left\langle e^{\lambda\mathcal{G}(0)+\lambda'\mathcal{G}(t)}\right\rangle$$

Improved two-point functions

- **Two-point function as:** $C(t) = \langle \mathcal{G}(t)\mathcal{G}(0)\rangle_c = \langle \mathcal{G}(t)\mathcal{G}(0)\rangle - \langle \mathcal{G}(t)\rangle\langle \mathcal{G}(0)\rangle$

finite-difference second derivative →

$$C(t) \simeq \frac{1}{\lambda\lambda'}\left[\left\langle \hat{w}_{\lambda,\lambda'}\right\rangle - \left\langle \hat{w}_{\lambda,0}\right\rangle\left\langle \hat{w}_{0,\lambda'}\right\rangle\right]$$

$$\left\langle \hat{w}_{\lambda,\lambda'}\right\rangle = \left\langle e^{\lambda\mathcal{G}(0)+\lambda'\mathcal{G}(t)}\right\rangle$$

- ☑ **Perfect flow limit exists: can obtain noiseless N-point functions**
- ☑ **In practice, significant variance reduction can be achieved**

Improved two-point functions

- **Two-point function as:** $C(t) = \langle \mathcal{G}(t)\mathcal{G}(0)\rangle_c = \langle \mathcal{G}(t)\mathcal{G}(0)\rangle - \langle \mathcal{G}(t)\rangle\langle \mathcal{G}(0)\rangle$

finite-difference second derivative →

$$C(t) \simeq \frac{1}{\lambda\lambda'}\left[\left\langle \hat{w}_{\lambda,\lambda'}\right\rangle - \left\langle \hat{w}_{\lambda,0}\right\rangle \left\langle \hat{w}_{0,\lambda'}\right\rangle\right]$$

$$\langle \hat{w}_{\lambda,\lambda'}\rangle = \langle e^{\lambda\mathcal{G}(0)+\lambda'\mathcal{G}(t)}\rangle$$

- ☑ **Perfect flow limit exists: can obtain noiseless N-point functions**
- ☑ **In practice, significant variance reduction can be achieved**

- **Practical implementation via "factorized flow" approximation: single flow used for several insertions**

$$f_{\lambda,\lambda'}(U) \simeq T^{-1}(t,0) \circ f_{\lambda'} \circ T(t,0) \circ f_{\lambda}(U).$$

$$\text{double flow} \simeq \text{translated flow} \circ \text{flow}$$

  - **Interpretation: local fluctuations dominate the variance**

Residual architecture for gauge theories

Architecture: residual layers

- Use an equivariant flow architecture based on the original Trivializing Maps [Lüscher, arXiv:0907.5491]
  [Abbott et al, 2305.02402]
  See also: [Bacchio et al, 2212.08469]
  [Gerdes et al, 2410.13161]
  [Nagai, Tomiya, 2103.11965]

$$U'_\mu(x) = e^{F(U)} U_\mu(x)$$

$$F = \sum_i \delta_i \, \mathrm{P}\left(W^i{}_{\mu\nu}\right)$$

Untraced Wilson loops that start and end at x

trainable

Traceless-antihermitian projection

Architecture: residual layers

- Use an equivariant flow architecture based on the original Trivializing Maps [Lüscher, arXiv:0907.5491]
  [Abbott et al, 2305.02402]
  See also: [Bacchio et al, 2212.08469]
  [Gerdes et al, 2410.13161]
  [Nagai, Tomiya, 2103.11965]

$$U'_\mu(x) = e^{F(U)} U_\mu(x)$$

$$F = \sum_i \delta_i \, \mathrm{P}\left(W^i{}_{\mu\nu}\right)$$

Untraced Wilson loops that start and end at x

trainable

Traceless-antihermitian projection

- Split lattice in active + frozen variables, and update only active (upper triangular Jacobian)

Architecture: residual layers

- Use an equivariant flow architecture based on the original Trivializing Maps [Lüscher, arXiv:0907.5491]
  [Abbott et al, 2305.02402]
  See also: [Bacchio et al, 2212.08469]
  [Gerdes et al, 2410.13161]
  [Nagai, Tomiya, 2103.11965]

$$U'_\mu(x) = e^{F(U)} U_\mu(x)$$

$$F = \sum_i \delta_i \, \mathrm{P}\left(W^i{}_{\mu\nu}\right)$$

Untraced Wilson loops that start and end at x

trainable

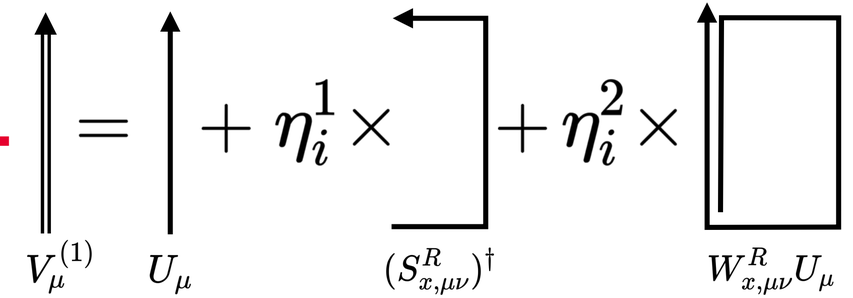
Traceless-antihermitian projection

- Split lattice in active + frozen variables, and update only active (upper triangular Jacobian)

- Build arbitrary loops "convoluting" the frozen links

$F(V)$

Force built from convoluted links



[Similar to L-CNN, Favoni et al, 2012.12901]

Connection to gradient flow

- **Gradient flow minimizes the action on a gauge configuration by solving the differential equation**
  [M. Lüscher, arXiv:1006.4518]

$$\dot{B}_\mu = -\frac{\delta S(U)}{dB_\mu} \longrightarrow U_\mu^{(i+1)} = e^{\epsilon F(U^{(i)})} U_\mu^{(i)}$$

solve numerically with infinitesimal steps

Connection to gradient flow

- **Gradient flow minimizes the action on a gauge configuration by solving the differential equation**
  [M. Lüscher, arXiv:1006.4518]

$$\dot{B}_\mu = -\frac{\delta S(U)}{dB_\mu} \longrightarrow U_\mu^{(i+1)} = e^{\epsilon F(U^{(i)})} U_\mu^{(i)}$$

solve numerically with infinitesimal steps

- **The residual layers act as a single and finite step of "generalized" gradient flow**

$$U'_\mu(x) = e^{F(U)} U_\mu(x)$$

$$F = \epsilon \times \sum_{\mu \neq \nu} \mathrm{P}(W_{\mu\nu}^{1\times 1})$$

Connection to gradient flow

- **Gradient flow minimizes the action on a gauge configuration by solving the differential equation**
  [M. Lüscher, arXiv:1006.4518]

$$\dot{B}_\mu = -\frac{\delta S(U)}{dB_\mu} \longrightarrow U_\mu^{(i+1)} = e^{\epsilon F(U^{(i)})} U_\mu^{(i)}$$

solve numerically with infinitesimal steps

- **The residual layers act as a single and finite step of "generalized" gradient flow**

$$U'_\mu(x) = e^{F(U)} U_\mu(x) \qquad F = \epsilon \times \sum_{\mu \neq \nu} \mathrm{P}(W_{\mu\nu}^{1\times 1})$$

- **Close to the original trivializing map proposal** **[M. Lüscher, arXiv:0907.5491]**

$$\log J = -\frac{4}{3}\epsilon \sum_{\mu\neq\nu} \mathrm{tr}\, W_{\mu\nu}^{1\times 1} + O(\epsilon^2) \longrightarrow \text{Qualitatively induces a change in the gauge coupling} \quad \beta \longrightarrow \beta + \Delta\beta$$

Timeslice operators

- Consider an interpolating operator on a fixed time slice $\mathcal{O}(t) = \sum_{\vec{x}} \mathcal{O}(\vec{x}, t)$
- Action includes operator at a fixed timeslice:

insertion at a fixed t

$$S_\lambda = S_{\mathrm{QCD}} + \lambda \mathcal{O}(t = 0)$$

Timeslice operators

- Consider an interpolating operator on a fixed time slice $\mathcal{O}(t) = \sum_{\vec{x}} \mathcal{O}(\vec{x}, t)$
- Action includes operator at a fixed timeslice:

insertion at a fixed t

$$S_\lambda = S_{\mathrm{QCD}} + \lambda \mathcal{O}(t = 0)$$

- Residual layers need to be modified to add time-dependent coefficients

"Force" in layers depends on Euclidean time

$$U'_\mu\left(\vec{x}, t\right) = e^{F(U|t,\mu)} U_\mu\left(\vec{x}, t\right)$$

- Volume transfer of models needs to account for broken symmetry
  - Train at target temporal extent, and transfer only in spatial directions, e.g. $4^3 \times 32$
  - No noticeable degradation with volume transfer [Abbott et al, 2603.02984]

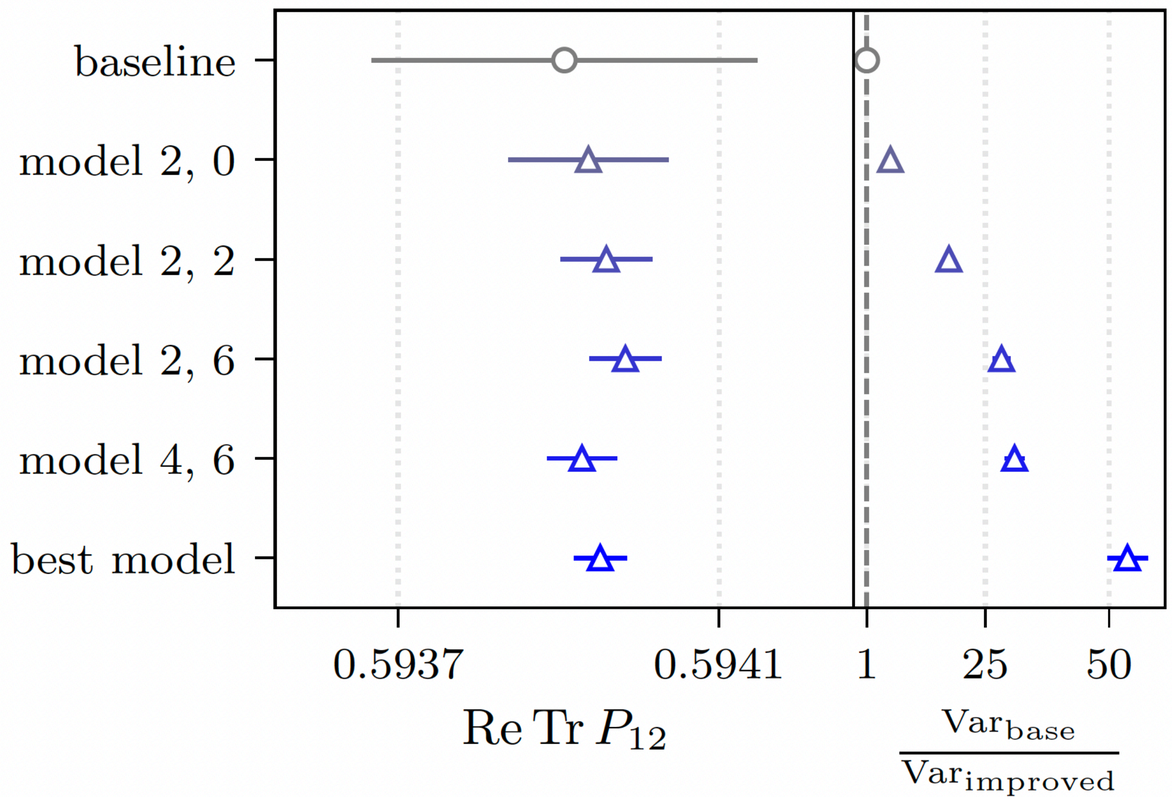
Numerical application: Loops & glueballs

Variance-reduced plaquette

- Consider a fixed-timeslice plaquette operator $\mathcal{G}(t) = \sum_{\vec{x}} P_{12}(\vec{x}, t)$ $\quad S_{\mathrm{YM}} \to S_{\mathrm{YM}} + \lambda \mathcal{O}$
  - Significant variance reduction in one-point function at fixed number of configurations

Increasing model complexity

(More convolution, Sparser masking pattern)

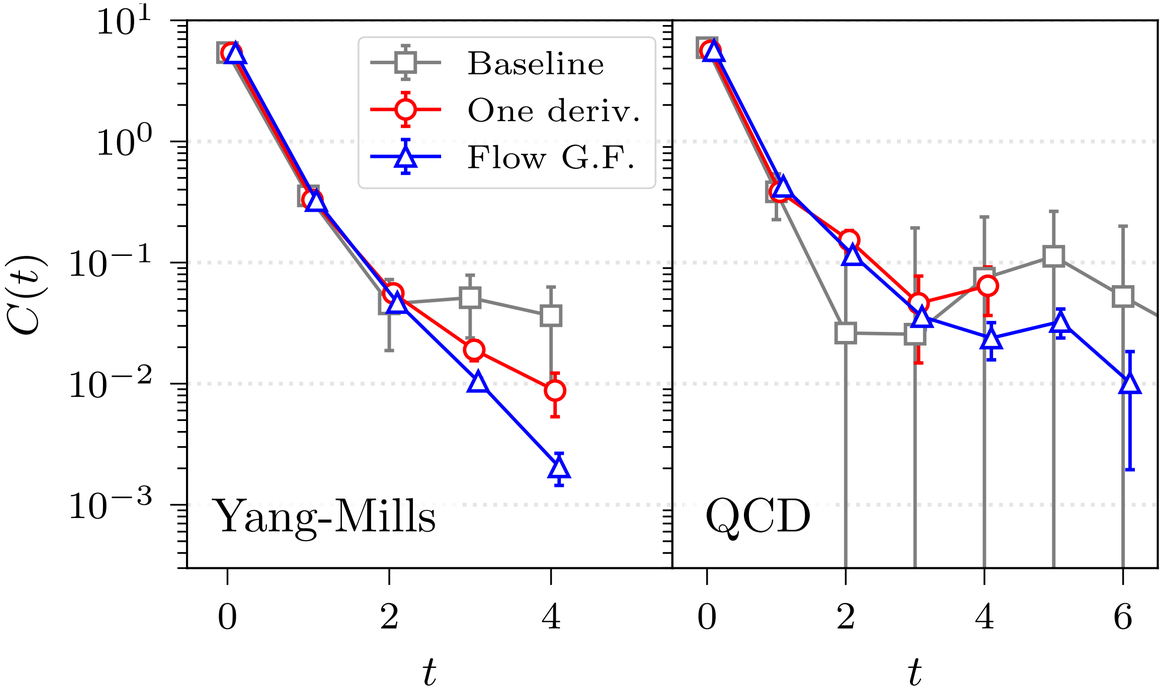


Increasing variance reduction

Glueballs in Yang-Mills & QCD

$$C(t) = \langle \mathcal{G}(t)\mathcal{G}(0)\rangle_c = \langle \mathcal{G}(t)\mathcal{G}(0)\rangle - \langle \mathcal{G}(t)\rangle\langle \mathcal{G}(0)\rangle \qquad \mathcal{G}(t) = \sum_{\vec{x}} P_{12}(\vec{x}, t)$$

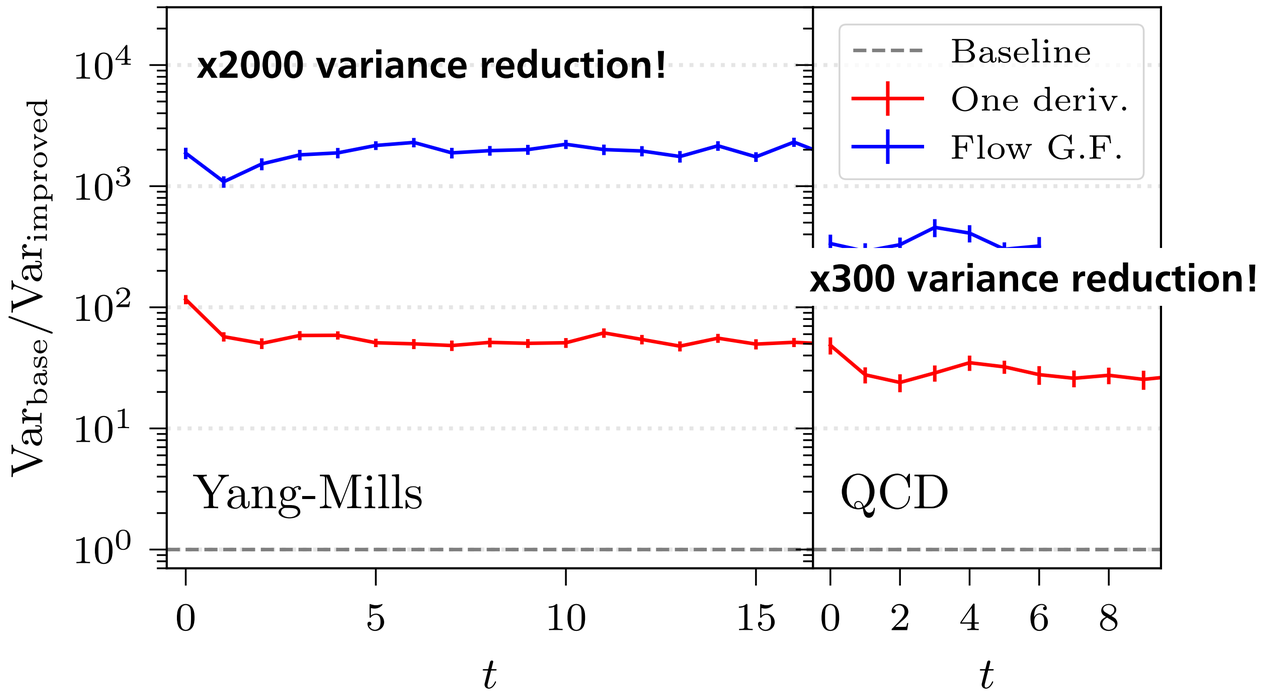
- **Correlator ($V = 16^3 \times 32$)**



$\beta = 6.0$ $\qquad$ $\beta = 5.7,\ \kappa = 0.1577$

$aM_\pi \sim 0.25,\ M_\pi \sim 600\,\mathrm{MeV}$

- **Variance reduction**



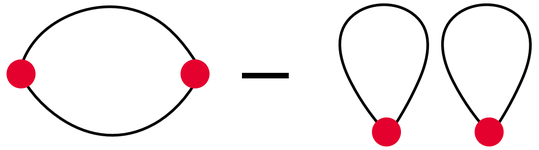
Numerical application: σ meson

The σ meson

- The sigma meson is a difficult target for hadron spectroscopy
  [2304.03762 (HadSpec) , 1908.01847 (GWQCD)]

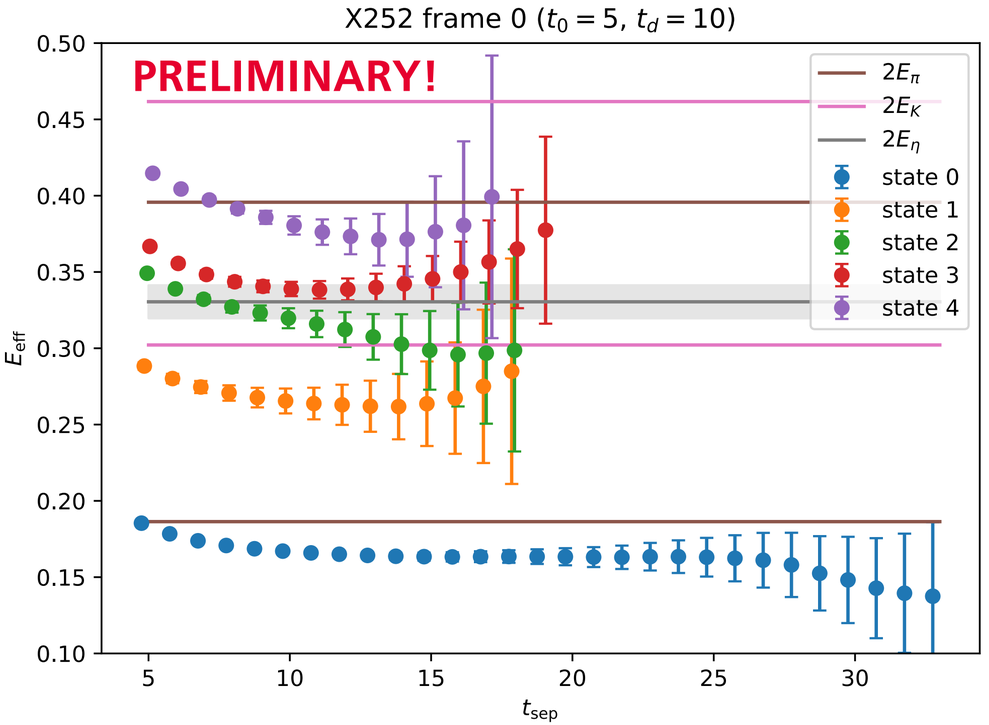
- Standard GEVP studies include correlators of the form

$$C(t) = \langle \sigma^\dagger(t)\, \sigma(0) \rangle =$$



$$\sigma \equiv \bar{u}u + \bar{d}d$$

- Noise is constant, while signal decays $\sim e^{-m_\sigma t}$



[Plot from Miguel Salg (Bern)]

Learning the σ meson

- Insert a sigma-meson operator at a fixed timeslice $$S_f \to S_f + \lambda_0 \sum_{\vec{x}} \bar{q}(\vec{x},0) q(\vec{x},0)$$
- This can be implemented by modifying the Dirac operator $$D_\lambda = D + \lambda_0 \delta_{\vec{x}\vec{y}}\, \delta_{t,0} \delta_{t',0}$$

Learning the σ meson

- Insert a sigma-meson operator at a fixed timeslice $$S_f \to S_f + \lambda_0 \sum_{\vec{x}} \bar{q}(\vec{x},0) q(\vec{x},0)$$
- This can be implemented by modifying the Dirac operator $$D_\lambda = D + \lambda_0 \delta_{\vec{x}\vec{y}}\, \delta_{t,0} \delta_{t',0}$$
- Use factorized flow approximation to compute two-point function

$$C(t) = \langle \sigma(t)\sigma(0)\rangle - \langle \sigma(t)\rangle\langle\sigma(0)\rangle \simeq \frac{1}{\lambda_0\lambda_t}\Big(\langle w_{\lambda_0\lambda_t}\rangle - \langle w_{\lambda_0}\rangle\langle w_{\lambda_t}\rangle\Big)$$

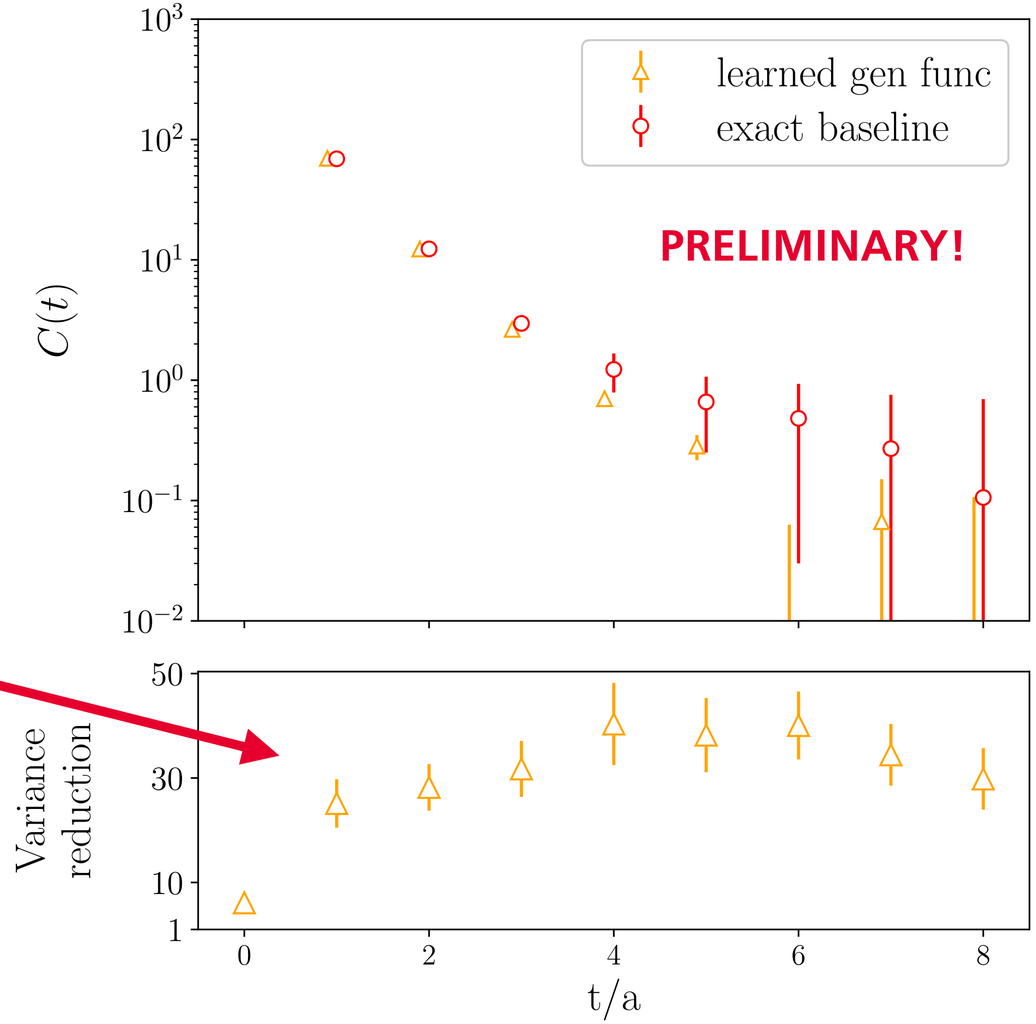
- For now, use small volume and evaluate fermion determinant and propagators exactly.

$$S_{\mathrm{QCD}}(U) = S_g(U) - \log\det D_\lambda[U]^\dagger D_\lambda[U],$$

$\mathrm{QCD\ with}\ N_f = 2\ \mathrm{Wilson\ fermions}$

$\beta = 5.7,\ \kappa = 0.1577,\ 4^3 \times 16$

Improved <σ σ> correlator



QCD with $N_f = 2$ Wilson fermions

$\beta = 5.7,\ \kappa = 0.1577$

$aM_\pi \sim 0.25,\ M_\pi \sim 600\,\mathrm{MeV}$

$4^3 \times 16$

(Exact evaluation of propagators and fermion determinants)

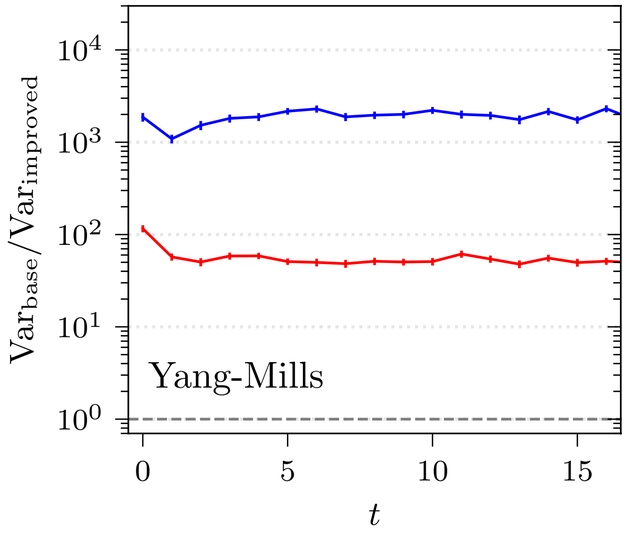
Higher variance reduction at longer time separation

Factor ~30x variance reduction!

Summary & outlook

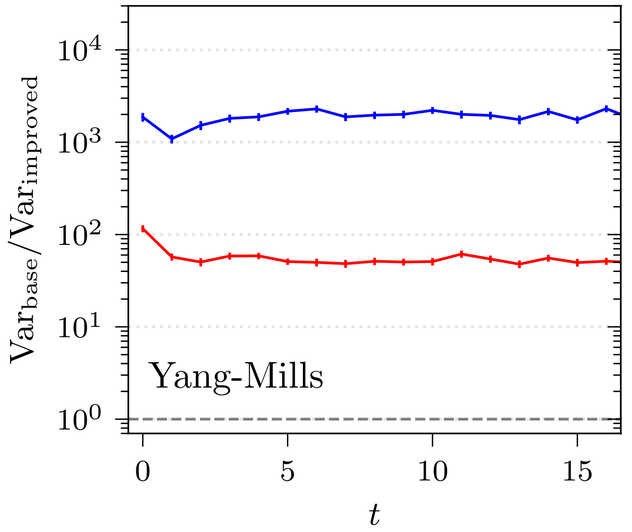
Summary and outlook

- ☑ The generating functional provides a natural framework for variance reduction
- ☑ Flows learn the insertion of operators: zero-variance limit is possible!
- ☑ Examples with > O(100) variance reduction: glueballs
- ☑ Promising preliminary results in meson correlators: σ meson
- ○ Ongoing training for glueball correlators in QCD with $M_\pi \sim 170$ MeV
  - Already achieved variance reduction: >100
- ○ Next steps: more observables and state-of-the-art QCD scales!
  - More complicated observables: smeared Wilson loops, other quark bilinears



Summary and outlook

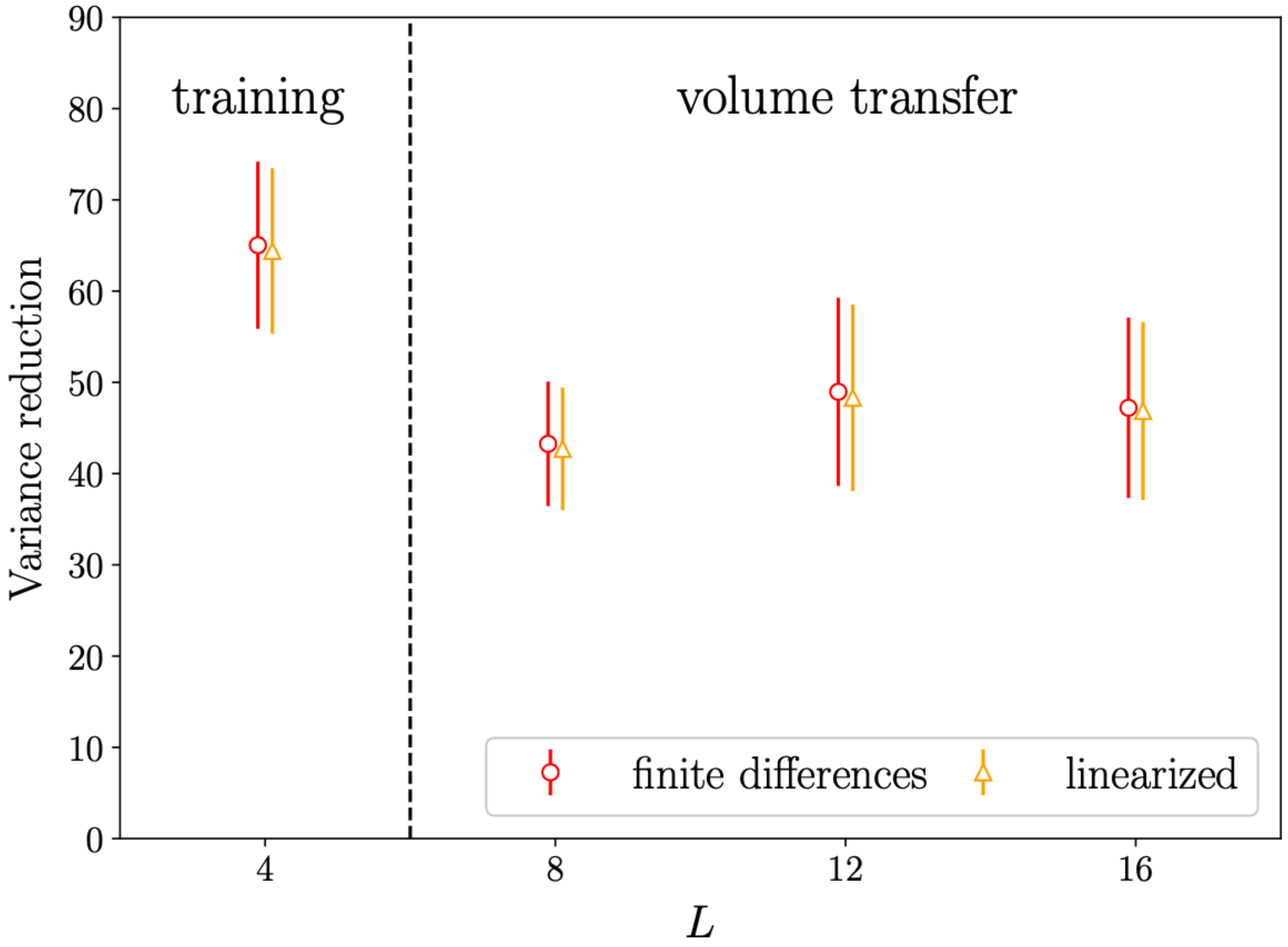
- ☑ The generating functional provides a natural framework for variance reduction
- ☑ Flows learn the insertion of operators: zero-variance limit is possible!
- ☑ Examples with > O(100) variance reduction: glueballs
- ☑ Promising preliminary results in meson correlators: σ meson
- ○ Ongoing training for glueball correlators in QCD with $M_\pi \sim 170$ MeV
  - Already achieved variance reduction: >100
- ○ Next steps: more observables and state-of-the-art QCD scales!
  - More complicated observables: smeared Wilson loops, other quark bilinears



Thanks!

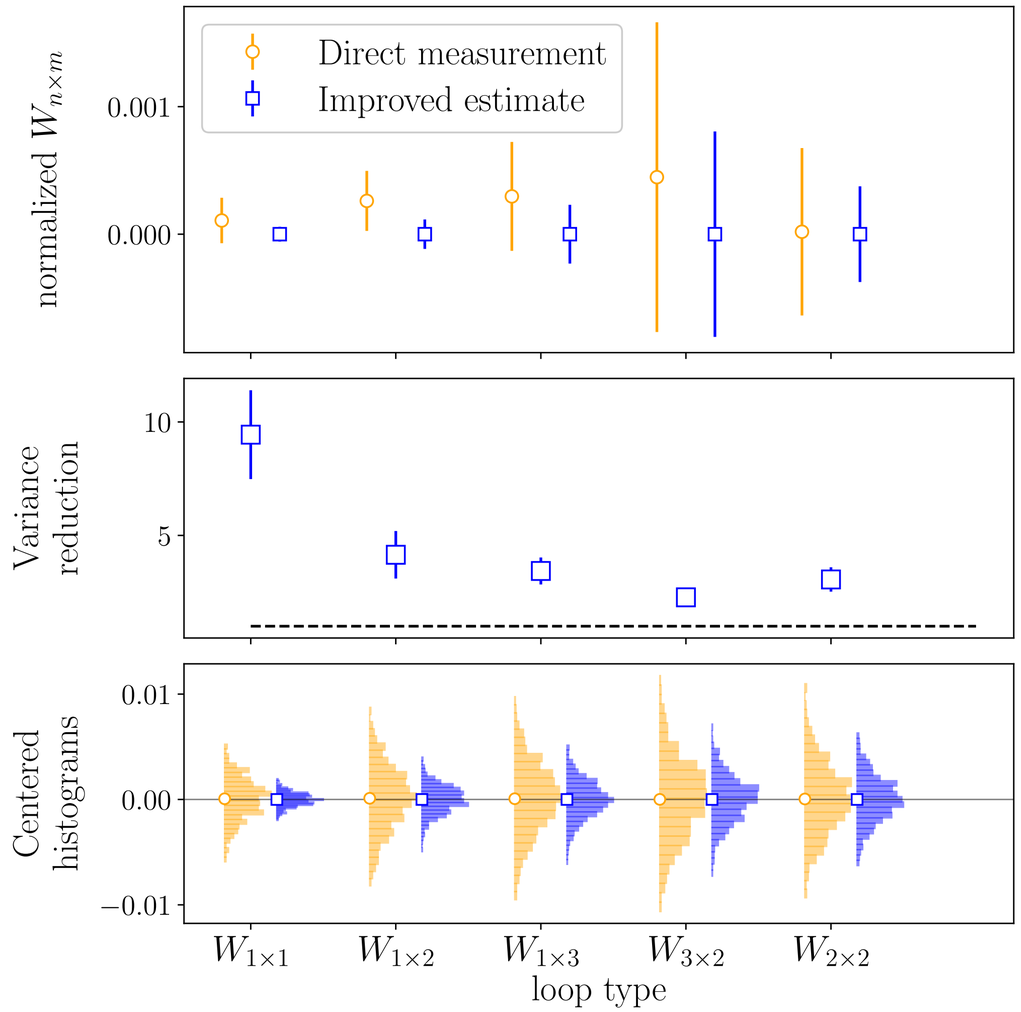
Back-up slides

Volume transfer



Other Wilson Loops

- ☑ **Variance reduction works for larger loops**
- **However, the gain is reduced**
  - **Larger loops are more nonlocal**
  - **Need different architectures?**



Using symmetry

