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]

[1]*Physics Department, Columbia University, New York, New York 10027, USA*
[2]*Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*
[3]*The NSF AI Institute for Artificial Intelligence and Fundamental Interactions*
[4]*Fermi National Accelerator Laboratory, Batavia, Illinois 60510, USA*
[5]*Higgs Centre for Theoretical Physics, School of Physics and Astronomy, University of Edinburgh, EH9 3FD Edinburgh, United Kingdom*
[6]*Albert Einstein Center, Institute for Theoretical Physics, University of Bern, 3012 Bern, Switzerland*

(Received 8 April 2026; accepted 1 June 2026; published 21 July 2026)

Normalizing flows can be used to construct unbiased, reduced-variance estimators for lattice field theory observables that are defined by a derivative with respect to action parameters. This work implements the approach for observables involving gluonic operator insertions in the SU(3) Yang-Mills theory and two-flavor QCD in four space-time dimensions. Variance reduction by factors of 10–60 is achieved in glueball correlation functions and in gluonic matrix elements related to hadron structure, with demonstrated computational advantages. The observed variance reduction is found to be approximately independent of the lattice volume, so volume transfer can be utilized to minimize training costs.

DOI: 10.1103/5xns-7qmj

I. INTRODUCTION

Lattice field theory provides a powerful numerical framework for computing physical observables in QCD, the fundamental theory of the strong interaction. Lattice QCD studies span from precision determinations of fundamental parameters of the Standard Model, to properties of nuclear systems. The approach is, however, notoriously computationally demanding: calculations require both the generation of gauge-field configurations via sophisticated sampling algorithms and the subsequent evaluation of observables through expensive measurement procedures. Improvements in either the sampling algorithms or the measurement methods can directly increase the scope and precision of lattice QCD [1].

Machine-learned normalizing flows have been proposed as tools to address both challenges [2]. From the perspective of sampling, using flows for gauge-field generation, replacing or supplementing more conventional Hybrid Monte Carlo approaches, has been shown to alleviate critical slowing down and improve topological sampling [3–5]. From the perspective of measurement, recent work [6–8] has proposed using flows to reduce the variance of observables that can be written as derivatives with respect to action parameters, with the first demonstration in QCD reported in Ref. [9]. This methodology utilizes flows to compute numerical derivatives by bridging between nearby distributions. While progress on both fronts remains rapid, applications to date have been demonstrated at scales and/or in regions of parameter space considerably different from the state-of-the-art regime of lattice QCD. Specifically, there have not yet been results demonstrating significant computational advantage for the setting of greatest phenomenological interest: calculations using large lattice volumes, for a variety of observables, in QCD with dynamical fermions.

This work demonstrates an application of flows that achieves significant variance reduction in several phenomenologically relevant applications in both dynamical QCD and Yang-Mills (YM) theories in four space-time dimensions. The study implements residual-flow architectures, introduced in Refs. [8,10], and is deployed for the first time for lattice geometries up to $16^3 \times 32$. Two categories of observables are investigated: 1) glueball correlation functions and 2) gluonic matrix elements in the context of hadron structure. In both cases the relevant observables can be expressed as derivatives with respect to action parameters. Variance reduction by a factor of 10–60 is observed across all cases. The implications of this reduction for the cost and workflow of lattice QCD calculations is discussed in detail. Importantly, the variance reduction does not

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI. Funded by SCOAP³.

degrade with volume transfer, which enables efficient training by volume-transfer approaches.

Addressing the bias of finite-difference approximations in derivative observables, this work also introduces a method to linearize learned flow transformations, yielding instantaneous flow fields that can be used to exactly evaluate the derivative with respect to action parameters. It is demonstrated that this unbiased estimate is equivalent to the control variate method of Refs. [6,7], in this case using the efficient masked residual-flow architecture and optimizing using the reverse Kullback-Leibler divergence as an objective function. It is proven and demonstrated numerically that, at leading order in the finite-difference parameter, the unbiased linearized estimator is identical to the finite-difference estimator on each gauge configuration. These estimates therefore reproduce the same variance reduction when flows are trained for short-distance transformations. At leading order, training short-distance residual flows with the reverse Kullback-Leibler (KL) divergence is equivalent to training flow fields directly using a physics-informed neural network (PINN) loss function [11], suggesting a future workflow in which both training and evaluation are obtained in this exact limit.

This work is organized as follows. First, the notion of derivative observables and how flows can be applied to reduce their variance is introduced in Sec. II. Section III summarizes the proposed workflow for such calculations. Section IV presents applications to glueball correlation functions, while Sec. V demonstrates applications for hadron structure. In Sec. VI, the potential computational advantage of the methodology is discussed. Section VII concludes and presents an outlook on future applications. The connection between the KL divergence and PINN loss is detailed in Appendix.

II. FLOWS FOR VARIANCE REDUCTION

This section describes the notion of derivative observables and how flow models can be used to reduce their variance.

A. Derivative observables

Many observables of interest in lattice QCD can be obtained by studying the derivative of path-integral quantities with respect to the action parameters. Two distinct classes of such derivative observables are considered in this work.

1. Vacuum-subtracted correlation functions via the derivative trick

A broad class of N-point correlation functions can be constructed from $(N-1)$-point functions by inserting an operator $\mathcal{Q}$ into the action and applying the "derivative trick" as follows. Assuming an action of the form $S_\lambda = S_0 - \lambda\mathcal{Q}$, where S_0 is the unperturbed action (in this case that of Yang-Mills theory or QCD), expectation values at a given value of λ are defined as

$$\langle\mathcal{O}\rangle_\lambda \equiv \frac{1}{\mathcal{Z}_\lambda}\int DU\,\mathcal{O}(U)e^{-S_\lambda(U)},$$
$$\mathcal{Z}_\lambda \equiv \int \mathcal{D}U\,e^{-S_\lambda(U)}. \tag{1}$$

In the following, $\langle\cdot\rangle_0$ will be used to denote expectation values in the unperturbed ($\lambda = 0$) ensemble. When $\mathcal{O}$ is a product of $N-1$ localized operators, taking a derivative of $\langle\mathcal{O}\rangle$ with respect to λ yields a vacuum-subtracted N-point correlation function:

$$\langle\mathcal{O}\mathcal{Q}\rangle_0 - \langle\mathcal{O}\rangle_0\langle\mathcal{Q}\rangle_0 = \left.\frac{d\langle\mathcal{O}\rangle_\lambda}{d\lambda}\right|_{\lambda=0}. \tag{2}$$

This is a specific application of the generating functional approach in quantum field theory, which, for example, also provides a route to computing susceptibilities and response functions from derivatives of expectation values.

2. Hadronic matrix elements via the Feynman-Hellmann theorem

A second application of the same underlying idea allows the extraction of hadronic matrix elements (see, e.g., Ref. [12]). Again taking $S_\lambda = S_0 - \lambda\mathcal{Q}$, the Feynman-Hellmann theorem relates the matrix element of $\mathcal{Q}$ in a hadronic state $|h\rangle$ at $\lambda = 0$ to the λ-dependence of the hadron energy $E_h(\lambda)$, extracted from two-point correlation functions computed with the modified action,

$$\langle h|\mathcal{Q}|h\rangle - \langle\Omega|\mathcal{Q}|\Omega\rangle = 2E_h\left.\frac{\partial E_h(\lambda)}{\partial\lambda}\right|_{\lambda=0}, \tag{3}$$

where $|\Omega\rangle$ is the interacting vacuum state. Unlike the derivative trick, which involves differentiating the correlator directly, the Feynman-Hellmann approach requires first extracting the energy eigenvalue from the spectral decomposition of the two-point function, then differentiating with respect to λ. This difference in the order of operations—differentiating the extracted energy rather than the correlator itself—cleanly isolates the matrix element from changes in the overlap factors and is a practical distinction between the two approaches.

B. Variance of derivative observables

The derivative trick provides an interesting perspective into the variance of derivative observables [7,13], or equivalently, why N-point functions are more noisy than $(N-1)$-point functions. This can be seen by writing out the derivative trick as an explicit limit of a finite difference,

$$\left.\frac{d\langle\mathcal{O}\rangle_\lambda}{d\lambda}\right|_{\lambda=0} = \lim_{\lambda\to 0}\frac{\langle\mathcal{O}\rangle_\lambda - \langle\mathcal{O}\rangle_0}{\lambda}. \tag{4}$$

The standard approach to estimating the N-point function is equivalent to evaluating $\langle\mathcal{O}\rangle_\lambda$ in the above expression by reweighting from expectation values under the $\lambda = 0$ action,

$$\begin{aligned}\langle\mathcal{O}\mathcal{Q}\rangle_0 - \langle\mathcal{O}\rangle_0\langle\mathcal{Q}\rangle_0 &= \lim_{\lambda\to 0}\left\langle \mathcal{O}\left(\frac{w_\lambda^{\mathrm{Id}} - 1}{\lambda}\right)\right\rangle_0 \\ &= \langle\mathcal{O}\partial_\lambda w_\lambda^{\mathrm{Id}}\rangle_0, \end{aligned} \tag{5}$$

where the normalized reweighting factor is

$$w_\lambda^{\mathrm{Id}}(U) = \frac{\mathcal{Z}_0}{\mathcal{Z}_\lambda}\hat{w}_\lambda^{\mathrm{Id}}(U) = \frac{\hat{w}_\lambda^{\mathrm{Id}}(U)}{\langle\hat{w}_\lambda^{\mathrm{Id}}\rangle_0} \tag{6}$$

in terms of the unnormalized weights

$$\hat{w}_\lambda^{\mathrm{Id}}(U) = e^{\lambda\mathcal{Q}(U)}. \tag{7}$$

Here the superscript "Id" denotes that this is a reweighting factor directly from S_0 to S_λ, without first applying a flow transformation to the field sampled from S_0, as is considered in future sections. The unsubtracted correlation function can be obtained by using the unnormalized weighting factors instead, as

$$\begin{aligned}\langle\mathcal{O}\mathcal{Q}\rangle_0 &= \lim_{\lambda\to 0}\left\langle \mathcal{O}\left(\frac{\hat{w}_\lambda^{\mathrm{Id}} - 1}{\lambda}\right)\right\rangle_0 \\ &= \langle\mathcal{O}\partial_\lambda\hat{w}_\lambda^{\mathrm{Id}}\rangle_0. \end{aligned} \tag{8}$$

Equations (5) and (8) can be confirmed by evaluating

$$\frac{\partial}{\partial\lambda}w_\lambda^{\mathrm{Id}}\bigg|_{\lambda=0} = \mathcal{Q} - \langle\mathcal{Q}\rangle_0, \qquad \frac{\partial}{\partial\lambda}\hat{w}_\lambda^{\mathrm{Id}}\bigg|_{\lambda=0} = \mathcal{Q}. \tag{9}$$

From the expression in Eq. (5), one sees that the subtracted N-point function obtained via the derivative trick can be expressed as the $(N-1)$-point observable reweighted by $\partial_\lambda w_\lambda^{\mathrm{Id}}$. Under the assumption that $\mathcal{O}$ and $\partial_\lambda w_\lambda^{\mathrm{Id}}$ are approximately independent,[1] the variance of the derivative observable can be approximated as

$$\mathrm{Var}\left[\frac{d\langle\mathcal{O}\rangle}{d\lambda}\right] \simeq \mathrm{Var}[\partial_\lambda w_\lambda^{\mathrm{Id}}]\langle\mathcal{O}^2\rangle + \cdots, \tag{10}$$

where the ellipsis indicates corrections due to correlations between the observable, its square, and the reweighting factors. In the finite-difference approximation, the relevant variance is

$$\mathcal{E}_{\mathrm{Id}}^2 := \mathrm{Var}\left[\frac{w^{\mathrm{Id}} - 1}{\lambda}\right], \tag{11}$$

for which $\lim_{\lambda\to 0}\mathcal{E}_{\mathrm{Id}}^2 = \mathrm{Var}[\partial_\lambda w_\lambda^{\mathrm{Id}}]$, so the variance of the derivative operator is proportional to $\mathcal{E}_{\mathrm{Id}}^2$ under the independence assumption. This conclusion provides an obvious path forward: modifying the reweighted estimate, for example, by incorporating flow transformations, to reduce fluctuations of the reweighting factor around 1 should lead to a lower variance in the observable.

C. Flows

A flow [14–16] is defined as a diffeomorphism f that maps samples from a base distribution $r(U)$ to a model distribution with density

$$q(V) = r(U)\times J(V), \qquad V = f(U), \tag{12}$$

where the Jacobian of the flow transformation is defined as

$$J(V) \equiv \left|\det\frac{\partial f(U)}{\partial U}\right|^{-1}. \tag{13}$$

The flow f is typically constructed with many free, trainable parameters, which are optimized such that the obtained model distribution approximates a target distribution p, i.e., $q(V)\simeq p(V)$.

In the context of sampling lattice QCD gauge fields [10], the samples U and V are lattice gauge-field configurations, and the probability distributions p and r are defined in terms of Euclidean lattice actions:

$$r(U)\propto e^{-S(U)}, \qquad p(V)\propto e^{-S'(V)}. \tag{14}$$

Here, S and S' represent the fact that the Euclidean actions have different parameters, and the proportionality is up to the normalization constant of the distribution. Any flow f can be used to construct unbiased estimates according to the target distribution by reweighting samples according to the weights $w(V) = p(V)/q(V)$, which can be explicitly computed using Eqs. (12) and (14).

There are several options to optimize the parameters defining the flow, all of which aim to minimize the difference between the model density $q(V)$ and the target density $p(V)$. For example, one can perform stochastic gradient descent on a direct estimate of the KL divergence [17], using reduced-noise estimates of the gradients with path gradients [18] or related methods [10], or avoiding gradients of $p(V)$ using the REINFORCE algorithm [19]. This work uses the path gradient approach to minimize the reverse KL divergence.

A common construction of expressive flows is based on the composition of invertible layers,

$$f = g_1\circ g_2\circ\ldots\circ g_n. \tag{15}$$

[1]Whether this is a good approximation depends strongly on the observable, and its validity cannot be established in general terms.

Several gauge-equivariant architectures for the invertible layers g_k were discussed in Ref. [10]. This work uses a modified version of the residual layers described in Ref. [8], which are based on the trivializing map [20]. Similar architectures were also used in Refs. [6,21–24].

Each layer transforms a subset of gauge links, defined as "active," conditioned on the remaining "frozen" links. The choice of partitioning is referred to as the masking pattern, which can be specified as $x_\mu \in \{0, 1\}$, where 0 and 1, respectively, denote frozen and active links at space-time position x in direction μ. In this work, "mod N" masking patterns are used, which satisfy the condition $(p + \sum_\mu x_\mu) = 0 \,(\mathrm{mod}\, N)$ for parities $p \in \{0, \ldots, N-1\}$. A "mod 2" masking pattern, for instance, corresponds to a checkerboard.

Active links in residual layer k are transformed as

$$U'_\mu(x) = e^{\lambda F^k_x(U_f, U_\mu(x))} U_\mu(x), \tag{16}$$

where the "flow field" of the layer $F^k_x(U_f, U_\mu(x))$ is formed from linear combinations of algebra-projected, untraced Wilson loops that begin and end at the position x. This construction ensures gauge equivariance, since under a gauge transformation with local parameter Ω_x the field transforms as $F^k_x \to \Omega_x F^k_x \Omega^\dagger_x$. Here, an explicit factor of λ, which defines the target, is included based on the empirical observation that inductive bias results in residual flow transformations that tend to scale as $O(\lambda)$, for small λ. This fact is used to efficiently linearize these flows in Sec. II F.

The coefficients used to build the flow field out of untraced Wilson loops are learnable. Schematically,

$$\lambda F^k_x(U_f, U_\mu(x)) = \sum_n \alpha_n \mathcal{P}[U_\mu(x) S^n_{x,\mu\nu}(U_f)], \tag{17}$$

where $\mathcal{P}[W]$ is a projection to anti-Hermitian, traceless elements of the algebra, $S^n_{x,\mu\nu}$ is a staple-shaped object that depends only on the frozen links, and α_n are trainable coefficients. In addition, a convolution is applied to the frozen links to increase the expressivity—see Ref. [8] for the precise implementation and Ref. [25] for a related architecture.

In the architecture presented in Ref. [8], translation invariance is used in the coefficients in the flow field; i.e., α_n in Eq. (17) are not position dependent. This increases training stability and reduces memory costs. However, some of the target distributions in this work have less symmetry, e.g., if a time-slice operator is inserted as in Eq. (57). To effectively learn such targets, a dependence on the time slice is introduced as $\alpha_n(x_0)$, explicitly breaking time-translation invariance in the parameters defining the flow field. This only leads to a moderate increase of training costs and memory usage.

It is also necessary to be able to evaluate the Jacobian determinant defining $J(V)$. The masking pattern applied to construct the transformation in Eq. (16) ensures that the Jacobian matrix is triangular with respect to the indices x. For the kth residual layer, only links in the μ direction are changed. The determinant of the Jacobian matrix of this layer is then entirely determined by the diagonal blocks and can be evaluated as

$$J_k(V) = \prod_{x \,\text{active}} |\det_{ab} \nabla^a_{x,\mu} \mathrm{Tr}[-2T^b(\lambda F^k_x)]|^{-1}. \tag{18}$$

Here, the symbol ∇ denotes the left Lie derivative, $\nabla^a_{x,\mu}\phi(U) = \frac{d}{dt}\phi(\ldots, e^{tT^a}U_\mu(x), \ldots)|_{t=0}$, and the generators are taken to be anti-Hermitian and normalized according to $\mathrm{Tr}[T^a T^b] = -\frac{1}{2}\delta^{ab}$. The full Jacobian is obtained by composing the Jacobian determinant of all layers. In this work, autodifferentiation is applied to evaluate the Jacobian factor for each active link.

D. Derivative observables via flows

A natural way to use flows in the computation of derivative operators is to consider the case where $S = S_0$ and $S' = S_\lambda = S_0 - \lambda\mathcal{Q}$. Finite-difference approximations of the derivatives appearing in the right-hand sides of Eqs. (2) and (3) may then be calculated by using flows to evaluate unbiased expectation values $\langle\cdot\rangle_\lambda$ and the corresponding unflowed fields to evaluate $\langle\cdot\rangle_0$. For the derivative trick, this leads to

$$\left.\frac{d\langle\mathcal{O}\rangle}{d\lambda}\right|_{\lambda=0} \simeq \frac{1}{\lambda}\langle w_\lambda(V)\mathcal{O}(V) - \mathcal{O}(U)\rangle_0, \tag{19}$$

where $V = f(U)$ is distributed according to $q(V)$, and the reweighting factor is defined as

$$w_\lambda(V) = \frac{\hat{w}_\lambda(V)}{\langle\hat{w}_\lambda(V)\rangle_q}, \qquad \hat{w}_\lambda(V) = e^{-S_\lambda(V)}/q(V). \tag{20}$$

In the Feynman-Hellmann application, the right-hand side of Eq. (3) may be similarly evaluated as

$$\left.\frac{\partial E_h}{\partial\lambda}\right|_{\lambda=0} \simeq \frac{1}{\lambda}(E_h(\lambda) - E_h(0)). \tag{21}$$

The perturbed energy $E_h(\lambda)$ is determined by fitting the standard spectral decomposition to the two-point hadronic correlation function $C^h_\lambda(t)$ computed on the flowed fields,

$$\begin{aligned} C^h_\lambda(t) &= \langle O^h_t(U)[O^h_0(U)]^\dagger\rangle_\lambda \\ &= \langle w_\lambda(V) O^h_t(V)[O^h_0(V)]^\dagger\rangle_0, \end{aligned} \tag{22}$$

where O^h_t and $[O^h_0]^\dagger$, respectively, indicate an annihilation operator at Euclidean time t and a creation operator at Euclidean time 0 with quantum numbers of the state $|h\rangle$. Evaluating this correlation function on the flowed fields

implies evaluating all gluonic quantities and all quark propagators using the flowed gauge configurations V. Section II F discusses an alternative to the finite-difference approximation which has particular advantages when computing correlation functions with multiple choices of the inserted operator.

The overall quality of a given flow model is often quantified with the effective sample size per configuration (ESS). In this context, it is defined as

$$\mathrm{ESS} = \frac{\langle w_\lambda \rangle_q^2}{\langle w_\lambda^2 \rangle_q} = \frac{1}{1 + \mathrm{Var}[w_\lambda]}. \tag{23}$$

The ESS can be estimated using N gauge field configurations generated from $q(V)$,

$$\mathrm{ESS} \approx \frac{1}{N} \frac{[\sum_{i=1}^N w_\lambda(V_i)]^2}{\sum_{i=1}^N [w_\lambda(V_i)]^2}, \tag{24}$$

with the value of this estimator lying in the interval $\mathrm{ESS} \in [\frac{1}{N}, 1]$. A perfect model corresponds to $\mathrm{Var}[w_\lambda] = 0$ and $\mathrm{ESS} = 1$.

In these applications, where a flow model is used to map between arbitrarily similar distributions (i.e., for infinitesimally small λ), the ESS can be arbitrarily close to 1, depending on the choice of λ itself. A more useful metric for model quality is then $\mathcal{E}^2$, as defined in Eq. (11) but for the trained flow, which can be related to the ESS according to

$$\mathcal{E}^2 = \frac{1}{\lambda^2} \left(\frac{1}{\mathrm{ESS}} - 1 \right) \tag{25}$$

and which approaches $\mathrm{Var}[\partial_\lambda w_\lambda]$ as $\lambda \to 0$. As a given model is trained, so that the ESS approaches 1, the result above shows that $\mathcal{E}^2 \to 0$, and an improvement in the variance of the estimated observable is expected, compared to the reweighting factor w_λ^{Id} in Eq. (5) which corresponds to the identity flow transformation, $f = \mathrm{Id}$. Under the independence assumption of Eq. (10), the ratio of $\mathcal{E}^2$ between the trained and identity flow can be used to predict the variance reduction for any measured derivative observables as

$$\frac{\mathrm{Var}[d\langle \mathcal{O} \rangle_{\mathrm{Id}}/d\lambda]}{\mathrm{Var}[d\langle \mathcal{O} \rangle/d\lambda]} \simeq \frac{\mathcal{E}_{\mathrm{Id}}^2}{\mathcal{E}^2}. \tag{26}$$

The Feynman-Hellmann application involves reweighted estimates of two-point functions as inputs to the spectral decomposition; thus, an analogous reduction in noise is expected from using flowed estimates of these two-point functions.

It is also possible to infer the expected volume scaling of the variance reduction. To see this, we note that the exponential volume scaling of the ESS [26] at sufficiently large volumes implies, for small λ,

$$\mathrm{ESS}(V) = 1 - kV\lambda^2 + O(\lambda^3), \tag{27}$$

where k is a constant. It then follows that

$$\mathcal{E}^2 = kV + O(\lambda), \tag{28}$$

so that $\mathcal{E}^2$ is extensive in the volume, and similarly for $\mathcal{E}_{\mathrm{Id}}^2$. Since the variance reduction can be approximated by the ratio appearing in Eq. (26), and both $\mathcal{E}^2$ and $\mathcal{E}_{\mathrm{Id}}^2$ exhibit the same volume scaling, the variance reduction is expected to be approximately independent of the volume.

E. Determinant ratio estimation

Several of the applications demonstrated in this work are evaluated for dynamical QCD. While the inserted operators used in these applications are purely gluonic, a ratio of fermion determinants must be estimated to determine the reweighting factors in Eq. (20). In this work, only the case of $N_f = 2$ degenerate light quark flavors are considered, in which case the $\lambda = 0$ action can be written in terms of the lattice gauge action S_g and the lattice Dirac operator D as

$$S_0(U) = S_g(U) - \log\det DD^\dagger(U). \tag{29}$$

The relevant reweighting factors are then

$$\begin{aligned} \log \hat{w}_\lambda(V) &= -S_\lambda(V) + S_0(U) - \log J(V) \\ &= \lambda \mathcal{Q}(V) - S_g(V) + S_g(U) \\ &\quad + \log\det \frac{DD^\dagger(V)}{DD^\dagger(U)} - \log J(V). \end{aligned} \tag{30}$$

To compute the fermionic contribution to the reweighting factor, the pseudofermion trick [27] can be used to write the reweighting factor as an expectation value under Gaussian samples ϕ,

$$\det \frac{DD^\dagger(V)}{DD^\dagger(U)} = \langle e^{-\phi^\dagger (MM^\dagger)^{-1}\phi} e^{\phi^\dagger \phi} \rangle_\phi, \tag{31}$$

where $M = [D(U)]^{-1}D(V)$ and $\langle \cdot \rangle_\phi$ indicates an average with respect to the Gaussian distribution $\exp(-\phi^\dagger\phi)$. This corresponds to the methodology of using "naive pseudofermions," which can lead to high variance in the reweighting factors. There are several ways to improve on this, as described in Ref. [28]: one can use even/odd preconditioning [29,30], include more pseudofermion hits and average the reweighting factors, or use a pseudofermion flow.

In this work, pseudofermion flows are used. For this, the gauge-equivariant "Parallel Transport Convolutional Network" (PTCN) architecture discussed in Ref. [28] is

applied. In this architecture, a Gaussian pseudofermion field is transformed iteratively by taking linear combinations with its parallel-transported neighbors. Schematically,

$$\phi'(x) = A\phi(x) + BU_\mu(x)\phi(x+\hat{\mu}) + CU_\mu^\dagger(x-\hat{\mu})\phi(x-\hat{\mu}) + \cdots, \tag{32}$$

where A, B, C represent matrices of trainable parameters. By dividing the pseudofermion field into active and frozen degrees of freedom, a tractable Jacobian $J(\phi')$ can be computed. Furthermore, the PTCN architecture can easily be combined with even-odd preconditioning; this approach is employed throughout this work. Given a trained pseudofermion flow, an improved estimate for the determinant ratio can then be obtained by evaluating

$$\det\frac{DD^\dagger(V)}{DD^\dagger(U)} = \langle e^{-\phi'^\dagger(MM^\dagger)^{-1}\phi'} e^{\phi^\dagger\phi} J(\phi')^{-1}\rangle_\phi. \tag{33}$$

F. Linearization of the models

Although the value of λ can be made small to control the $O(\lambda)$ bias in the derivative [Eqs. (19) and (21)], a more elegant and robust approach is to eliminate the bias entirely. This can be achieved through an exact linearization of the approach. An additional advantage is that the gauge fields are not changed and quark propagators do not need to be recomputed for each operator insertion, as will be discussed below.

In the limit $\lambda \to 0$, the optimal flow $f(U)$ approaches the identity transformation plus an $O(\lambda)$ linear term. The behavior of the flow in this limit can be expressed in terms of the instantaneous *flow field* $F(U)$ as

$$V = \{1 + \lambda F(U)\}U + O(\lambda^2) = e^{\lambda F(U)}U + O(\lambda^2). \tag{34}$$

Above, the flow field $F(U)$ is a field of anti-Hermitian, traceless elements of the group algebra describing the tangent direction of the flow expanded around the original gauge configuration U.

In this limit, it is therefore possible to exactly evaluate the derivative in λ by expanding to first order. Expanding the weight first gives

$$\hat{w}_\lambda(V) = \frac{e^{-S_0(V)+\lambda\mathcal{Q}}}{e^{-S_0(U)}J(V)} = 1 + \lambda[\nabla\cdot F - F\cdot\nabla S_0 + \mathcal{Q}] + O(\lambda^2), \tag{35}$$

where all terms in the expansion are evaluated on the gauge configuration U. The operator ∇ here indicates a vector of operators $\nabla^a_{x,\mu}$, so

$$F\cdot\nabla \equiv \sum_{\mu,x,a} F^a_{x,\mu}\nabla^a_{x,\mu}, \qquad \nabla\cdot F \equiv \sum_{\mu,x,a}(\nabla^a_{x,\mu}F^a_{x,\mu}). \tag{36}$$

The normalized weight introduces an extra factor that gives the vacuum subtraction,

$$w_\lambda(V) = \frac{Z_0}{Z_\lambda}\hat{w}_\lambda = (1 - \lambda\langle\mathcal{Q}\rangle + O(\lambda^2))\hat{w}_\lambda = 1 + \lambda\underbrace{[\nabla\cdot F - F\cdot\nabla S_0 + \mathcal{Q} - \langle\mathcal{Q}\rangle]}_{\partial_\lambda w_\lambda} + O(\lambda^2). \tag{37}$$

The observable can likewise be expanded as

$$\mathcal{O}(V) = \mathcal{O}(U) + \lambda[F(U)\cdot\nabla\mathcal{O}(U)] + O(\lambda^2). \tag{38}$$

Combining these terms, an unbiased estimator (i.e., with no finite-difference approximation) for the λ derivative of an arbitrary observable is finally given by [cf. Eq. (19)]

$$\left.\frac{d\langle\mathcal{O}\rangle}{d\lambda}\right|_{\lambda=0} = \langle(\partial_\lambda w_\lambda)\mathcal{O} + F\cdot\nabla\mathcal{O}\rangle_0 = \langle K\mathcal{O}\rangle_0 + \langle\mathcal{Q}\mathcal{O}\rangle_0 - \langle\mathcal{Q}\rangle_0\langle\mathcal{O}\rangle_0, \tag{39}$$

where the linear operator

$$K = \nabla\cdot F - F\cdot\nabla S_0 + F\cdot\nabla \tag{40}$$

encodes the effect of the flowed estimate of the derivative. The correctness of Eq. (39) follows because K acts as a total derivative when taken in an expectation value with any smooth test function,

$$\langle K\mathcal{O}\rangle_0 = \frac{1}{\mathcal{Z}_0}\int \mathcal{D}U\,\nabla\cdot(F\mathcal{O}e^{-S_0(U)}) = 0. \tag{41}$$

For N-point functions evaluated using the derivative trick, the variance improvement obtained by the flowed definition of the derivative can be viewed in this light as the introduction of a particularly motivated control variate $K\mathcal{O}$, which vanishes in expectation but which should anticorrelate with $\mathcal{Q}\mathcal{O}$ to reduce fluctuations of the estimator [6]. Optimizing the flow drives $w_\lambda \to 1$, so the linear term of the weight must approach zero,

$$\partial_\lambda w_\lambda = \nabla\cdot F - F\cdot\nabla S_0 + \mathcal{Q} - \langle\mathcal{Q}\rangle \approx 0. \tag{42}$$

In this limit, the dependence on $\mathcal{Q}$ in Eq. (39) approximately disappears, and the derivative evaluation reduces to an $(N-1)$-point function,

$$\frac{d\langle\mathcal{O}\rangle}{d\lambda} \approx \langle F\cdot\nabla\mathcal{O}\rangle_0. \tag{43}$$

Correspondingly, for weights approximately independent of the observable, the prefactor $\mathrm{Var}[\partial_\lambda w_\lambda]$ determining the noise of the correlation function is driven to zero.

This perspective also motivates a future training approach based on a PINN loss function that directly minimizes Eq. (42); optimizing in this way is shown in Appendix to be equivalent, at leading order in λ, to minimizing the KL divergence for a short-distance flow.

Similarly, for Feynman-Hellmann applications, the exact derivative of the two-point function [cf. Eq. (22)] can be computed as

$$\left.\frac{dC^h(t)}{d\lambda}\right|_{\lambda=0} = \langle (K + \mathcal{Q} - \langle \mathcal{Q} \rangle_0) O_t^h [O_0^h]^\dagger \rangle_0. \quad (44)$$

The matrix element of interest [cf. Eq. (3)] may then be obtained from the time dependence of the ratio

$$\frac{dC^h(t)/d\lambda|_{\lambda=0}}{C_0^h(t)} \xrightarrow{t\gg 1} \text{const} + t \left.\frac{dE_h}{d\lambda}\right|_{\lambda=0}, \quad (45)$$

whose slope at large t yields $dE_h/d\lambda$. This is the standard summed-insertion method: because $\mathcal{Q}$ is inserted on every time slice, the sum over intermediate times produces the characteristic linear growth in t that projects onto the ground-state matrix element.

In practice, the flow field associated with this infinitesimal limit can be obtained from a trained finite flow by evaluating the full flow transformation $V = f(U)$ and approximating

$$F(U) \approx \frac{1}{\lambda} \mathcal{P}[V U^\dagger]. \quad (46)$$

However, in this work, the residual-flow architecture used provides an efficient shortcut to define a flow field $F(U)$ from a trained flow. In this case, expanding the composition of residual layers that defines V in Eq. (46) gives up to $O(\lambda)$,

$$V = U + \lambda \sum_k F_x^k(U) + O(\lambda^2), \quad (47)$$

in terms of the flow fields F_x^k defined for each layer in Eq. (16). Then, Eq. (46) leads to the approximation

$$F(U) = \sum_k F_x^k(U). \quad (48)$$

In this case, the exact divergence $\nabla \cdot F(U)$ can likewise be evaluated using the known Jacobian of the residual flow transformation.

As discussed above, we empirically find that individual flow fields $F_x^k(U)$ take $O(1)$ values after training, which justifies the expansion in Eq. (47). While in principle one could imagine other parametric λ-dependence appearing in the layers of the trained model, in practice this appears not to be preferred by the inductive bias of the parametrization used here.

Finally, as for finite flows, an estimator involving the Dirac operator is required to evaluate the term $F \cdot \nabla S_0 = F \cdot \nabla S_g - F \cdot \nabla \log\det DD^\dagger$ for dynamical QCD applications. The latter term can be evaluated using either naive or flowed pseudofermions similarly to the finite flow case,

$$\begin{aligned} F \cdot \nabla \log\det DD^\dagger &= \mathrm{Tr}[(D^\dagger)^{-1}(F \cdot \nabla DD^\dagger) D^{-1}] \\ &= \langle (D^{-1}\phi)^\dagger (F \cdot \nabla DD^\dagger) D^{-1}\phi \rangle_\phi \\ &= \langle (D^{-1}\phi')^\dagger (F \cdot \nabla DD^\dagger) D^{-1}\phi' J(\phi')^{-1} \rangle_\phi. \end{aligned} \quad (49)$$

This is equivalent to the $O(\lambda)$ expansion of Eq. (33) used for the case of a finite flow, so the noise properties of this estimate are equivalent for small λ.

III. SUMMARY OF THE WORKFLOW

Below, a concrete workflow that may be used to train and evaluate derivative-trick observables is detailed for both the finite-difference approximation and the unbiased linearized estimator. This approach is used for the numerical applications discussed in Secs. IV and V.

A. Workflow for flowed derivative observables

a. Using a finite-difference approximation. To estimate a derivative observable using a flow trained for a finite value of λ, the following workflow is used:

(1) Train a gauge-field flow f at small lattice volume. In dynamical QCD, the small volume allows an exact evaluation of the fermion determinant to be used.
(2) For dynamical QCD, train a pseudofermion flow at small volume.
(3) Apply flow $V_i = f(U_i)$ for N gauge configurations U_i at the target volume. Evaluate the corresponding weight $\hat{w}_\lambda$, either exactly (for pure-gauge YM) or using the flow-improved pseudofermion estimator of the determinant ratio (for QCD).
(4) Evaluate the derivative observable as

$$\frac{1}{N} \sum_i \left[\frac{\hat{w}_\lambda(V_i)}{\frac{1}{N}\sum_j \hat{w}_\lambda(V_j)} \mathcal{O}(V_i) - \mathcal{O}(U_i) \right]. \quad (50)$$

b. Using linearization. To estimate a derivative observable using a linearized flow, providing an unbiased estimate of the $\lambda \to 0$ limit, the following workflow is used:

(1) Train a gauge-field flow f at small volume. Extract the linearized flow field F.
(2) For dynamical QCD, train a pseudofermion flow at small volume.
(3) Evaluate flow field $F_i = F(U_i)$ for N gauge configurations U_i at target volume. Evaluate the corresponding weight derivative

$$\partial_\lambda \hat{w}_\lambda = \nabla \cdot F_i - F_i \cdot \nabla S_0(U_i) + \mathcal{Q}(U_i), \quad (51)$$

either exactly (for pure-gauge YM) or using the pseudofermion estimator defined above (for dynamical QCD).

(4) Evaluate the unbiased derivative observable as

$$\frac{1}{N}\sum_i [(\partial_\lambda \hat{w}_{\lambda,i})\mathcal{O}(U_i) + F_i \cdot \nabla \mathcal{O}(U_i)] - \left(\frac{1}{N}\sum_i \partial_\lambda \hat{w}_{\lambda,i}\right)\left(\frac{1}{N}\sum_j \mathcal{O}(U_j)\right). \tag{52}$$

B. Cost comparison between linearization and finite differences

In addition to the obvious advantage of eliminating $O(\lambda)$ effects in the linearized approach, there are other practical considerations. Notably, the finite-difference approach requires applying the full flow and evaluating observables and reweighting factors on the flowed gauge fields. In contrast, the linearized method only requires evaluating the observables and weights using the original gauge-field ensemble $\{U_1, U_2, \ldots\}$. This can be important when working with fermionic observables because Dirac operator inversions only need to be computed on the original, unflowed gauge configurations.

Another issue to consider is the total number of inversions required in each workflow. As an explicit example, consider the case of evaluating the meson three-point functions

$$C_i^{(3)}(x) = \langle \mathcal{Q}_i(\bar{\psi}_x \Gamma \psi_x)(\bar{\psi}_0 \Gamma' \psi_0)\rangle \tag{53}$$

for n choices of operators $\mathcal{Q}_1, \ldots, \mathcal{Q}_n$, where Γ and Γ' represent arbitrary Dirac gamma structures.

Following the finite-difference method, each three-point function can be obtained by applying the derivative trick to the two-point observable $\mathcal{O} = \langle(\bar{\psi}_x \Gamma \psi_x)(\bar{\psi}_0 \Gamma' \psi_0)\rangle_\psi$ with distinct flows $f_1, \ldots, f_n$ that have been respectively trained to target each inserted operator. Evaluating these n correlation functions with finite flows requires evaluating

$$\mathcal{O}(V) = \mathrm{Tr}[D_{0x}^{-1}(V)\Gamma D_{x0}^{-1}(V)\Gamma'] \tag{54}$$

by constructing the propagator[2] $D_{x0}^{-1}(V)$ on the flowed fields V for each flow, as well as for the unflowed fields—a total of $n+1$ propagators.

In contrast, evaluating the same correlation functions by linearizing the flows requires calculating the term

$$F \cdot \nabla \mathcal{O} = -\sum_{z,z'}\{\mathrm{Tr}[D_{0z}^{-1}(F \cdot \nabla D_{zz'})D_{z'x}^{-1}\Gamma D_{x0}^{-1}\Gamma'] + \mathrm{Tr}[D_{0x}^{-1}\Gamma D_{xz'}^{-1}(F \cdot \nabla D_{z'z})D_{z0}^{-1}\Gamma']\}, \tag{55}$$

which can be computed following two different methodologies. First, one can use sequential inversion through the term $F \cdot \nabla D$. In this case, evaluating for each flow field $F \in \{F_1, \ldots, F_n\}$ requires $n+1$ propagator inversions to access all sink times; i.e., the counting is identical to the finite-difference case. Second, one can use sequential inversion through the sink. In this case, the flowed correlation function can be computed for an arbitrary number of target operators by contracting with the respective flow fields $F_i \cdot \nabla D$, requiring $n_{\mathrm{sink}} + 1$ propagator inversions, where n_{sink} is the number of sink times. For a large number of operators, as is typically the case for insertions involving quark bilinears with different momenta, the latter approach is advantageous even if several sink times are used. These two approaches precisely echo the standard choice of inverting through the operator or sink to compute fermionic three-point functions.

This highlights the benefit of the linearized approach in this context: three-point functions can be evaluated using the same propagator inversions that would be used for the standard evaluation of three-point functions involving quark bilinear operators, while maintaining the variance-reduction provided by the flowed approach. This significantly reduces the inversions needed when a large number of operator choices are considered and allows us to reuse any propagators that would already be calculated on the target ensemble for analogous three-point function calculations. Despite this, for cases in which only a few operator insertions are considered, or for practical convenience in avoiding sequential inversions, a finite-difference calculation with controlled bias may be preferable to the linearized flow estimates. As such, in the remainder of this work, both finite and linearized flow estimates are investigated.

IV. NUMERICAL APPLICATIONS TO GLUEBALL SPECTROSCOPY

This section presents applications of the methodology developed here to reduce the variance of glueball correlators. Specifically, correlators of the glueball with quantum numbers $J^{PC} = 0^{++}$ (i.e., the scalar glueball) are studied in both SU(3) Yang-Mills theory and QCD in four space-time dimensions.

Using the derivative trick in Eq. (2), the scalar glueball two-point function is connected to the derivative of a one-point function,

$$C(t) = \langle \mathcal{O}(t)\mathcal{Q}\rangle_0 - \langle \mathcal{O}(t)\rangle_0\langle \mathcal{Q}\rangle_0 = \left.\frac{d\langle \mathcal{O}(t)\rangle}{d\lambda}\right|_{\lambda=0}, \tag{56}$$

with $S_\lambda = S_0 - \lambda\mathcal{Q}$ as in Sec. II. The interpolating operators used in the above definition are chosen to be

$$\mathcal{O}(t) = \sum_{\vec{x}}\sum_{i\neq j=1}^{3} \mathrm{Re}\,\mathrm{Tr}\,P_{ij}(\vec{x}, t), \qquad \mathcal{Q} = O(0), \tag{57}$$

[2]The transpose is obtained by γ^5-hermiticity.

TABLE I. Details of the trained models used for the demonstration of glueball correlator computation. For Model B, the two rows of evaluation metrics correspond to evaluations with naive pseudofermions ("B") and with a pseudofermion flow model ("B + PF").

Model	Theory	Parameters	Train geometry	Train ESS/$\mathcal{E}^2$	Evaluation geometry	Evaluation ESS/$\mathcal{E}^2$
A	SU(3) YM	$\beta = 6.0,\ \lambda = 2 \times 10^{-3}$	$4^3 \times 32$	99.996%/10	$16^3 \times 32$	99.6%/950
B	$N_f = 2$ QCD	$\beta = 5.6, \kappa = 0.1577, \lambda = 2 \times 10^{-3}$	$4^3 \times 32$	99.996%/10	$16^3 \times 32$	98.5%/3800
B + PF						98.8%/3000

with $P_{ij}(x)$ the (untraced) 1×1 Wilson loop along the spatial i, j directions. The operator insertion has been localized on a particular time slice, which ensures that the spectral decomposition of the resulting two point functions holds. To take advantage of time-translation averaging, the derivative-trick evaluations of $C(t)$ are repeated for all possible temporal translations of each gauge configuration. The effect this has on the overall computational cost is discussed in Sec. VI.

The derivative definition of the vacuum-subtracted correlation function holds both in the pure gauge theory ($S_0 = S_{\mathrm{YM}}$) and in QCD ($S_0 = S_{\mathrm{QCD}}$). Below, both applications are shown. A summary of models used is presented in Table I.

A. Glueballs in Yang-Mills theory

The scalar glueball is first investigated in the SU(3) Yang-Mills theory. A flow model is constructed and trained as described in Sec. II. The architecture is defined as two stacks[3] of residual layers, the first with masking pattern "mod 2" and the second "mod 4," as described in Ref. [8]. The target distribution is defined by fixing $\lambda = 10^{-3}$. All models were trained using a $4^3 \times 32$ lattice geometry, then transferred to the target lattice volumes. Model details and resulting quality metrics after training are reported as "model A" in Table I. The model was trained for approximately 10,000 optimizer steps, corresponding, in a non-optimized implementation, to about 3000 NVIDIA A100 GPUhours.

The results for the correlation function evaluated at the largest target lattice geometry of $16^3 \times 32$ are shown in Fig. 1. Here, both the finite-difference and linearized strategies outlined in Sec. III are compared against a baseline result obtained using the standard translation-averaged, vacuum-subtracted evaluation of the two-point function in Eq. (56). The correlation functions computed using the flow model with both approaches are consistent with the baseline correlation functions within uncertainties for all time slices. However, the flow-improved estimates give a constant variance reduction of a factor of around 50 across all time slices, resulting in a measurable signal persisting for two additional time slices. The results for the finite-λ case are also indistinguishable from the linearized case in terms of both central value and variance reduction. This nontrivial check verifies the equivalence of the two approaches in the limit $\lambda \to 0$.

The ratio $\mathcal{E}^2_{\mathrm{Id}}/\mathcal{E}^2$ should provide the leading-order approximation to the observable-specific variance reduction, under the independence approximation in Eq. (10). For this example, at the evaluation volume, the identity flow has ESS = 89%, which leads to $\mathcal{E}^2_{\mathrm{Id}} = 3.1 \times 10^4$. The ratio is then $\mathcal{E}^2_{\mathrm{Id}}/\mathcal{E}^2 \simeq 32$, which is consistent with the order of magnitude of the observed variance reduction.

Another metric demonstrating the efficacy of the flow approach is the precision of the resulting effective mass estimates. As a representative example, the effective mass defined using the correlator at time slices 1 and 2 is calculated,

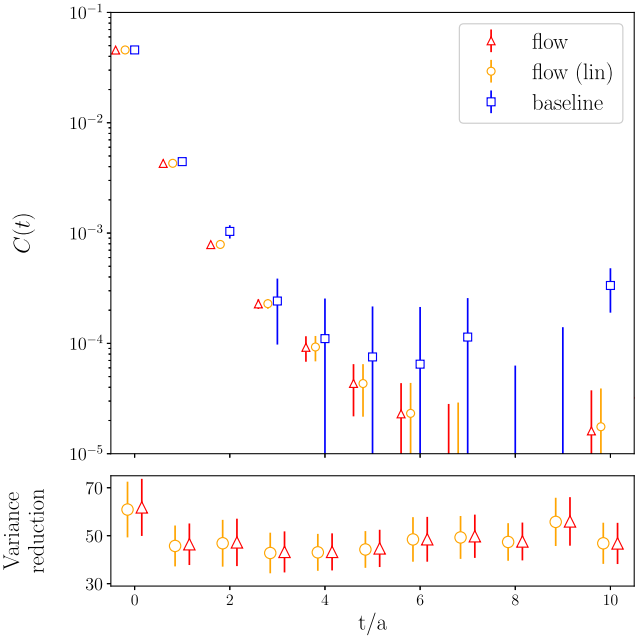


FIG. 1. Illustration of variance reduction in the glueball correlator in the SU(3) Yang-Mills theory with $\beta = 6$ for a $16^3 \times 32$ lattice geometry. The upper panel displays the correlator itself, evaluated on an ensemble of 3200 configurations. The baseline calculation using the standard two-point function evaluation is compared against the derivative-trick estimate evaluated using both the finite flow with $\lambda = 2 \times 10^{-3}$ and the unbiased linearized estimate. In the lower panel, the ratio between the variance of the baseline and flow-improved estimates is shown.

[3]A "stack" of layers is a set of layers such that every link is actively transformed once, composed.

$$am_{\mathrm{eff}} = -\log[C(2a)/C(a)], \tag{58}$$

giving the flow-improved and baseline values

$$am_{\mathrm{eff}}(\mathrm{flow}) = 1.695(23),$$
$$am_{\mathrm{eff}}(\mathrm{baseline}) = 1.48(15). \tag{59}$$

Both results are consistent within 2σ, and the variance in the flow case is approximately a factor of 40 smaller. While this observable is still affected by excited state effects, and therefore does not directly estimate the glueball mass, it is used to emphasize that the observed variance reduction is not isolated to the correlator and persists in derived quantities.

Next, the hypothesis that the reweighting factors drive the variance of the correlation function can be tested by comparing the variance of the flowed estimates with and without the inclusion of the corresponding reweighting factors. For the glueball correlator studied here, the unweighted correlation function is found to be significantly more precise than the reweighted correlation function, albeit with a growing bias particularly at large values of t. This explains the behavior of the observed flow-improved correlation function at the present level of training: the flow successfully models the short distance correlation functions but mismodels the tail, requiring increasing contributions from noisy reweighting factors to provide the correct expectation value. This highlights an opportunity for further gains, as a more expressive flow with further training could reduce the deviation from the exact correlation function, requiring significantly smaller and thus more precise reweighting factors.

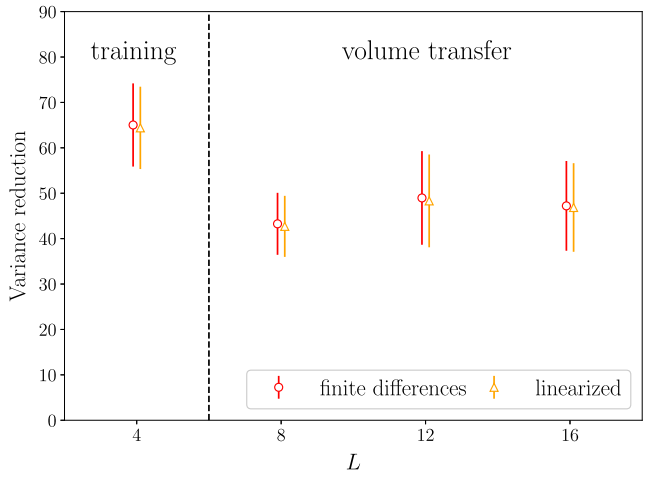


FIG. 2. Variance reduction in the glueball correlator achieved by the flow approach at time slice $t = 2a$ as a function of the $L^3 \times 32$ lattice geometry. Results for both the finite-λ and linearized approach are shown. Data for the $L = 16$ case is obtained from an ensemble of 3200 configurations while the remaining cases are evaluated using ensembles of approximately 1600 configurations each.

Finally, the volume dependence of the flow-improved correlator is investigated by evaluating the trained flow across additional lattice geometries. While it is well known that normalizing flows used for direct sampling deteriorate exponentially with increasing volume [26], in Sec. II D, we have argued why this is not expected for the variance reduction application. Indeed, the numerical results here also confirm this. In Fig. 2, the variance reduction for a fixed time slice is shown for calculations using lattice geometries $L^3 \times 32$ ranging from the training geometry with $L = 4$ to the largest target geometry with $L = 16$. The net gain achieved by a fixed flow—measured by the variance reduction factor—is approximately constant with volume. The results are nearly identical for both the finite-λ and linearized approaches. This demonstrates the viability of training at small volumes, then transferring to evaluate the resulting models at significantly larger volumes. This strategy minimizes training costs with minimal loss in performance.

B. Glueballs in QCD

The scalar glueball correlation function is next investigated in QCD. In this case, the same interpolating operators defined in Eq. (57) are used to define the correlation function, but the underlying theory is taken to be QCD with $N_f = 2$ Wilson fermions. The action parameters are set to $\beta = 5.6$ and $\kappa = 0.1577$, corresponding to $aM_\pi = 0.2392(11)$, $M_\pi L \simeq 3.8$, $a \simeq 0.077$ fm and $M_\pi \simeq 600$ MeV.

In this setting, training the flow model requires significantly more computation per training step than in the Yang-Mills example, because inversions of the Dirac operator are needed to generate training samples with Hybrid Monte Carlo. This is mitigated first by working again with a small lattice volume for the training step. The cost is further controlled by exploiting model transfer, which reduces the total number of training iterations until convergence. In particular, the trained model from Sec. IV A is used as the initial model, from which only 1000 additional training steps are applied with the dynamical QCD target action to reach convergence. To reduce the noise in the gradients, this training is performed using the explicit determinant of the Dirac operator. While this is feasible in terms of memory use for the training volume ($4^3 \times 32$), it also incurs additional computational costs. Overall, with these choices, the computational cost per gradient step during training is approximately 10 times larger than in the pure-gauge case, while the number of training steps is 10 times smaller, leading to a roughly equivalent training cost.

In contrast to the pure gauge case, the computation of reweighting factors in the evaluation volume involves a pseudofermion estimate of the determinant ratio, as explained in Sec. II E. This typically leads to a lower ESS in the evaluation volume than an exact determinant-ratio estimator. Here, two strategies are compared: a naive

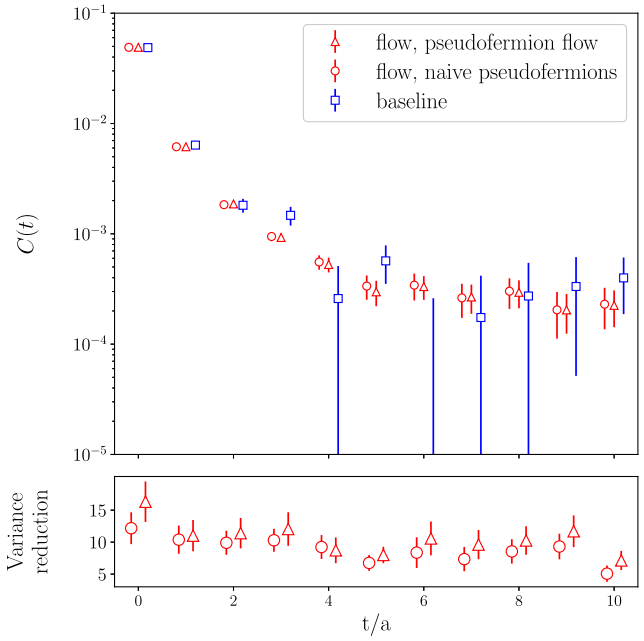


FIG. 3. Illustration of the variance reduction in the glueball correlator in QCD with $N_f = 2$ Wilson fermions using 1280 configurations. The upper panel displays the correlator itself; the flow-improved estimates using both naive pseudofermions and a pseudofermion flow are compared against a baseline calculation using the standard computation of the two-point function. A single pseudofermion hit for both flow estimates. In the lower panel, the ratio of variances is shown with and without employing a pseudofermion flow.

pseudofermion estimator with even/odd preconditioning and a determinant-ratio estimator based on a pseudofermion flow model. The latter leads to a higher ESS with a fixed number of pseudofermion hits with a negligible overhead in evaluation costs, bringing the final ESS closer to the value that would be obtained by exactly evaluating the determinant ratio. The resulting model details and quality metrics after training are reported as "model B" (using naive pseudofermions) and "model B + PF" (using a pseudofermion flow model) in Table I.

The results are shown in Fig. 3, both with naive pseudofermions and with a pseudofermion flow model, using a single pseudofermion sample per gauge configuration in both cases. Even with naive pseudofermions, a significant variance reduction by a factor of 5–10 is obtained at the level of the correlation function, extending the signal region by one to two time slices. Including a pseudofermion flow model further reduces the variance by roughly 20%.

Computing the effective mass between $t = a$ and $t = 2a$, as in the Yang-Mills case, also demonstrates an improvement in precision. The effective mass estimates for the flow estimate with naive pseudofermions, the flow estimate with a pseudofermion flow model, and the baseline estimate are

$$\begin{aligned} am_{\mathrm{eff}}(\mathrm{flow}) &= 1.212(39), \\ am_{\mathrm{eff}}(\mathrm{flow,PF}) &= 1.193(33), \\ am_{\mathrm{eff}}(\mathrm{baseline}) &= 1.26(15). \end{aligned} \tag{60}$$

Relative to the baseline, the variance is reduced by a factor of 14 using with naive pseudofermions and 20 with a pseudofermion flow model.

V. NUMERICAL APPLICATION TO HADRON STRUCTURE

Another class of derivative operators of great phenomenological interest are hadronic matrix elements, which can be expressed in the form of derivatives through the Feynman-Hellmann theorem as in Eq. (3). This procedure assumes that the operator is inserted in every timeslice, for which time-translation invariance is preserved.

Here, this approach is applied to a study the gluon momentum fraction of the pion in dynamical QCD. A gluonic operator is used for this demonstration as this kind of operator suffers particularly severely from signal-to-noise challenges, and variance reduction on a finite ensemble may thus be particularly important in this case. The approach can also be extended to quark operators.

The gluon momentum fraction of the pion is computed following the methodology of Ref. [12]. Specifically, the operator considered is a linear combination of components of the discretized energy-momentum tensor (EMT),

$$\mathcal{Q}_{\mathrm{gEMT}} = -\frac{\beta}{N_c}\sum_x \mathrm{Tr}\,\mathrm{Re}\left(\sum_i P_{i0} - \sum_{i<j} P_{ij}\right), \tag{61}$$

where $i, j \in \{1, 2, 3\}$. The matrix element of this operator, obtained in the pion rest frame via Eq. (3), is related to the gluon momentum fraction of the pion $\langle x \rangle_g$ by

$$\left.\frac{dM_\pi}{d\lambda}\right|_{\lambda\to 0} = \frac{1}{2M_\pi}\langle \pi|\mathcal{Q}_{g\mathrm{EMT}}|\pi\rangle = -\frac{3M_\pi}{2}\langle x\rangle_g^{\mathrm{latt}}, \tag{62}$$

where "latt" indicates that it is a bare matrix element. The full action $S_\lambda = S_0 - \lambda\mathcal{Q}_{\mathrm{gEMT}}$ can be expressed as an anisotropic action, where temporal and spatial plaquettes have different couplings. This is an amenable target for the residual layers, as shown in Ref. [8].

Following this methodology, the gluon momentum fraction is determined in $N_f = 2$ QCD using twisted-mass fermions and the tree-level improved gauge action. This setup, chosen for simplicity and to reduce discretization effects, is described in Ref. [31], where the lattice spacing is determined to be approximately $a = 0.10$ fm. The parameters are $\kappa = 0.164111$, $\beta = 3.8$, and $\mu = 0.012$, yielding a pion mass of approximately $M_\pi \simeq 540$ MeV. The target distribution is defined by fixing $\lambda = 2 \times 10^{-3}$.

TABLE II. Details of the trained models used for the demonstration of hadronic matrix element calculation via the Feynman-Hellmann theorem. The two rows of evaluation metrics correspond to evaluations with naive pseudofermions ("C") and with a pseudofermion flow model ("C + PF").

Model	Theory	Parameters	Train geometry	Train ESS/$\mathcal{E}^2$	Evaluation geometry	Evaluation ESS/$\mathcal{E}^2$
C	QCD	$\kappa = 0.164111, \beta = 3.8, \mu = 0.012, \lambda = 2 \times 10^{-3}$	4^4	99.94%/150	$16^3 \times 32$	65.0%/1.3×10^5
C + PF						69.2%/1.1×10^5

All models were trained using a 4^4 lattice geometry, then transferred to a target lattice geometry of $16^3 \times 32$. Model details and resulting quality metrics after training are reported as "model C" (using naive pseudofermions) and "model $C + \mathrm{PF}$" (using a pseudofermion flow model) in Table II; in both cases, three pseudofermion samples are used to evaluate reweighting factors. The model was trained for approximately 6k optimizer steps, corresponding, in a non-optimized implementation, to about 4000 NVIDIA A100 GPUhours.

The standard two-point correlation function is constructed on the flowed and unflowed ensembles via Eq. (22). Fits in the range $[t_{\mathrm{min}}, T/2]$ yield the pion mass, the mass derivative is estimated from the finite difference between the results at $\lambda = 2 \times 10^{-3}$ and $\lambda = 0$, and the gluon momentum fraction is then extracted by Eq. (62).

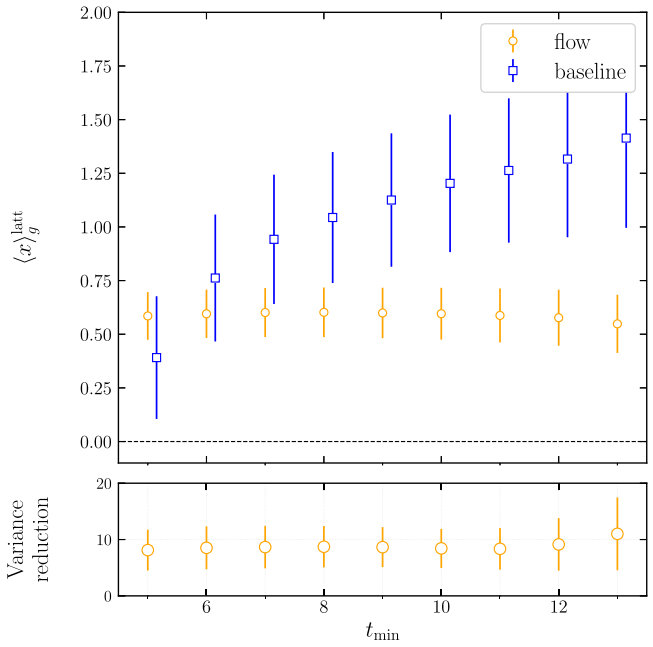


FIG. 4. Illustration of the variance reduction in the gluon momentum fraction in QCD with $N_f = 2$ Wilson fermions. The upper panel compares the gluon momentum fraction of the pion computed using flows (orange squares) versus the baseline ϵ-reweighting method (blue circles), both evaluated on the same ensemble of 8576 configurations. In the lower panel, the ratio between the variance of the baseline and the flow-improved estimate is shown. The results are shown as a function of the lower end of the fit range used for the extraction of the mass, t_{min}.

The results are shown in Fig. 4, where the gluon momentum fraction as a function of t_{min} is shown. To provide a baseline against which to measure variance improvement, results are compared to those computed by taking the derivative of the mass directly [32] (implemented in practice via ϵ-reweighting [8]). The variance of the momentum fraction estimate is reduced relative to the baseline by a factor of approximately 10 by using the flow approach. Moreover, the cost of the flow approach (using finite differences) is only slightly higher: while correlation functions need to be evaluated using the flowed fields, the inversions from each unflowed field can be used to initialize the solver on the corresponding flowed field to accelerate this calculation. As such, the cost of the flowed approach is less than a factor of 2 larger than the standard approach.

VI. DISCUSSION OF COMPUTATIONAL ADVANTAGE

The results of Secs. IV and V have demonstrated significant variance reduction in several observables computed on a fixed number of gauge fields, achieved by combining the derivative trick with application of flow models. However, using the flow methodology also increases the computational cost. Thus, assessing whether the demonstrated variance reduction represents true computational advantage requires considering all aspects of the computational pipeline, including the training and application of a flow model, configuration generation, measurement, and reweighting. Moreover, beyond sheer computational costs, there are other factors to consider: storage needs for configurations and propagators as well as data management of large numbers of measurements (which is particularly relevant for, e.g., studies of glueballs which may use a very large number of gauge fields).

A general feature of the examples considered in this work is that training costs are marginal: as a result of the small lattice volume used for training, these costs are smaller than generation of the target-volume configurations and for these examples are negligible in the computational budget. Moreover, even using a small training volume, training costs can be mitigated by transfer learning, as shown in the QCD glueball example, and simultaneous calculations of multiple observables can exploit this to amortize training costs even further. Rather than the training, it is thus the *application and reweighting* associated with the flow model that are potentially significant costs.

For the first example considered (glueballs in Yang-Mills theory), the cost of generating configurations via heatbath/overrelaxation and the cost of performing measurements and reweighting are low, as there are no fermions and so no inversions are necessary. The dominant cost in this case is applying the flow transformation, and therefore the flow approach is significantly more expensive than standard methods. This is in part due to the fact that the operator is inserted in each time slice, to take advantage of time-translation averaging, so the flow must be applied T times. While the variance reduction by a factor of roughly 50 is significant, this falls short of compensating for the additional computational overhead for the implementation used here by approximately a factor of 5. Presumably, additional investment in flow architectures and training could improve upon this to the vicinity of the break-even point.

Here, however, the primary advantage of using flows can be manifest in another aspect: calculations of glueballs in pure gauge theories tend to use very large ensembles, e.g., 10^7 gauge configurations in Ref. [33]. As such, a reduction of a factor of 50 in the number of configurations required can be valuable, even if computational costs break even. Fewer configurations lead to reduced storage costs, less data movement, and therefore lighter analysis pipelines, which can be of great significance at these data volumes.

This is precisely the advantage obtained in the calculation of glueballs in QCD. While the variance reduction achieved by using flows is in this case only a factor of 10, configuration generation costs are more significant. With the current flow model quality, the net result is approximately equal cost of flowed and unflowed approaches at fixed target error, with an order of magnitude fewer configurations required in the flow case. This demonstrates already a practical advantage, and further optimization of the flow models can yield both practical and computational gains.

Finally, the case of the hadron structure application in this work demonstrates clear computational advantage with the current flow model quality. Here, the flow evaluation and reweighting costs are subdominant for two primary reasons: the flow must only be applied once per configuration because of time-translation invariance, and propagators are computed on several sources per configuration to evaluate correlation functions. In this context, the cost of the finite-difference version of the flow pipeline is only marginally higher than the standard pipeline because the inversions on unflowed fields can be reused to initialize solves on the corresponding flowed fields; the additional inversion only adds a 30–50% overhead in cost. A similar cost is expected for the linearized evaluation, for which the inversions required are analogous to standard calculation of three-point functions. For a pipeline where the cost of HMC is dominant, reducing the variance by a factor of 10 directly yields a 10 times computational advantage. For a measurement-dominated pipeline, the advantage is approximately a factor of 7. For the particular implementations used in this demonstration, the advantage is approximately a factor of 8.

Taken in sum, the examples demonstrate that current flow technology can yield both computational and practical advantages in lattice QCD calculations. The precise magnitude of computational efficiency gain at a fixed target precision depends intimately on the details of the calculation and model used (as well as the efficiency of its implementation), but for the present implementation and applications, factors of a few have been demonstrated. In all cases, however, the flow approach yields an important practical advantage in that it allows for a variance reduction given a fixed-size ensemble of gauge fields. As ensembles are stored and reused as community resources, a fixed-size ensemble is in many cases typical, and improved variance in this setting represents a useful objective. This advantage is magnified toward the continuum limit and physical quark masses, where costs of ensemble generation are more significant.

VII. CONCLUSION

This work demonstrates the applicability of machine-learned flows as a tool to reduce the variance of lattice QCD calculations. This approach is based on formulating observables of interest as derivatives with respect to action parameters—a method that is applicable to arbitrary N-point functions using the generating functional approach. These calculations can then be interpreted as a reweighting problem between nearby distributions, where machine-learned flows are effective at reducing the variance of the reweighting factors by a learned change of variables. In this work, variance reductions by factors of 10–100 are demonstrated for a range of applications involving inserted gluonic operators, including glueball correlation functions and matrix elements in the context of hadron structure. An unbiased estimator for flowed derivative observables is also constructed by linearizing the machine-learned flow, removing a potential source of systematic error in the method. The generality of the approach, applied here for the first time for dynamical QCD quantities in four-dimensional lattice geometries up to $16^3 \times 32$, suggests that the flow-based variance reduction procedure is potentially useful for a wide range of applications in lattice QCD.

The computational advantage of this methodology has also been analyzed for each example studied. In the case of hadron structure, a lower uncertainty is obtained with the flow approach for the same number of measurements, resulting in a computational advantage of a factor of approximately 8. In the glueball case, the flow approach is comparable or slightly more expensive, but with clear practical advantages and the potential of additional computational gains with further refinement.

An important conclusion of this work is that the advantage achieved by utilizing flows for variance reduction does

not deteriorate when transferring to larger volumes. This was explicitly illustrated for the case of glueballs in Yang-Mills theory in Fig. 2, and it is expected to hold for other cases. As such, training can be performed at small volumes at very low computational cost that is negligible for the examples investigated here.

While this work studies dynamical QCD, all considered operator insertions are gluonic. An important future extension is therefore to explore fermionic operators, such as scalar or vector currents. This extension is conceptually straightforward but will require some new flow architecture development for the operator insertion. Similar success in this setting would yield variance reduction in observables such as sigma terms or the hadronic vacuum polarization tensor.

Ultimately, this work has demonstrated computational and practical advantages of a flow approach in the evaluation of various phenomenologically interesting observables in lattice QCD. This opens a new avenue of at-scale applications, in which machine-learned flows can now be included in the standard lattice QCD toolkit.

ACKNOWLEDGMENTS

We thank Michael Albergo, Kyle Cranmer, Mathis Gerdes, Alberto Ramos, and Ross Young for useful discussions. R. A. is an Ernest Kempton Adams Postdoctoral Fellow supported in part by the Ernest Kempton Adams fund for Physical Research of Columbia University and in part by U.S. DOE Grant No. DE-SC0011941. R. A., D. B., Y. F., P. E. S., and J. M. U. are supported in part by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics, under Contract No. DE-SC0011090. P. E. S. is additionally supported by the U.S. DOE Early Career Award No. DE-SC0021006, by a NEC research award, and by the Carl G and Shirley Sontheimer Research Fund. The work of F. R. L. was supported in part by the Platform for Advanced Scientific Computing (PASC) project "ALPENGLUE." This document was prepared using the resources of the Fermi National Accelerator Laboratory (Fermilab), a U.S. Department of Energy, Office of Science, Office of High Energy Physics HEP User Facility. Fermilab is managed by Fermi Forward Discovery Group, LLC, acting under Contract No. 89243024CSC000002. This work is supported by the U.S. National Science Foundation under Cooperative Agreement No. PHY-2019786 (the NSF AI Institute for Artificial Intelligence and Fundamental Interactions, http://iaifi.org/). It is also associated with an ALCF Aurora Early Science Program project and has used resources of the Argonne Leadership Computing Facility which is a DOE Office of Science User Facility supported under Contract No. DEAC02-06CH11357. The authors acknowledge the MIT SuperCloud and Lincoln Laboratory Supercomputing Center [34] for providing HPC resources that have contributed to the research results reported within this paper. Part of the calculations were performed on UBELIX (https://www.id.unibe.ch/hpc), the HPC cluster at the University of Bern. Numerical experiments and data analysis used PYTORCH [35], JAX [36], HAIKU [37], HOROVOD [38], NUMPY [39], and SCIPY [40]. Figures were produced using MATPLOTLIB [41].

DATA AVAILABILITY

There are no publicly available research data or software supporting this manuscript. Requests for further information or data should be sent to the authors.

APPENDIX: LOSS FUNCTIONS AND OPTIMIZATION

All models in this work are trained using the Adam optimizer applied to path gradient estimates of the reverse Kullback-Leibler divergence between the model density $q(V) = r(U) \times J(V)$ and the target density $p(V) \propto e^{-S_\lambda(V)}$.[4] Explicitly, the loss function and gradient estimates using the path gradient approach are defined in terms of expectation values with respect to $r(U) \propto e^{-S_0(U)}$ as

$$\begin{aligned}\mathcal{L}^{\mathrm{KL}} &= \langle \log J(V) + S_\lambda(V) - S_0(f^{-1}(V)) \rangle_0, \\ \nabla_\theta \mathcal{L}^{\mathrm{KL}} &= \langle (\nabla_\theta V) \cdot \nabla_V (\log J(V) + S_\lambda(V) - S_0(f^{-1}(V))) \rangle_0,\end{aligned} \tag{A1}$$

where $\nabla_\theta V \equiv P[\frac{\partial f(U)}{\partial \theta} f(U)^\dagger]$ indicates the algebra-valued gradient of $V = f(U)$ with respect to the parameters θ that define the flow f and ∇_V indicates the algebra-valued left Lie gradient with respect to the variable V. The dot product is here taken to be between the components in the algebra of $\nabla_\theta V$ and $\nabla_V(\ldots)$.

While the above loss function is used for all numerical applications in this work, a second loss function has also been investigated for directly training flow fields $F(U)$. The observation that $\partial_\lambda w_\lambda \approx 0$ should hold for an optimized flow [cf. Eq. (42)] suggests that a flow field could alternatively be trained by directly minimizing this quantity. This leads to a PINN loss function—one that aims to solve a partial differential equation by direct optimization—which is expressed as

$$\begin{aligned}\mathcal{L}^{\mathrm{PINN}} &= \langle (\partial_\lambda w_\lambda)^2 \rangle_0 \\ &= \langle (\nabla \cdot F - F \cdot \nabla S_0 + \mathcal{Q} - \langle \mathcal{Q} \rangle)^2 \rangle_0.\end{aligned} \tag{A2}$$

All quantities in the expectation value are evaluated for the fields U sampled from $r(U)$. An analogous loss function has previously been proposed for generative modeling via

[4]Here, we also assume that in the case of a dynamical theory, the action S_λ includes the effects of the fermions, either directly through the log fermion determinant or through a pseudofermion action. Thus, all results in this Appendix also hold for QCD.

annealing [11] and for directly optimizing control variates of lattice field theory observables [6].

Remarkably, this loss function is closely related to the KL divergence. Simplifying the gradient of the PINN loss,

$$\begin{aligned}\nabla_\theta \mathcal{L}^{\mathrm{PINN}} &= 2\langle \partial_\lambda w_\lambda \nabla_\theta (\nabla \cdot F - F \cdot \nabla S)\rangle_0 \\ &= -2\langle \nabla_\theta F \cdot (\nabla \partial_\lambda w_\lambda)\rangle_0, \end{aligned} \tag{A3}$$

where the second line is obtained by integrating by parts with respect to the gradient ∇. Expanding the gradient in Eq. (A1) with respect to λ, one finds that the gradients are related according to

$$\nabla_\theta \mathcal{L}^{\mathrm{PINN}} = \lim_{\lambda\to 0} \frac{2}{\lambda^2} \nabla_\theta \mathcal{L}^{\mathrm{KL}}; \tag{A4}$$

i.e., up to a rescaling by a factor of $\lambda^2/2$, the loss functions are equivalent to leading nonzero order. A slightly more involved derivation also shows that the loss functions themselves satisfy the same relationship. This suggests a future workflow in which *flow fields* $F(U)$ are directly constructed and trained, without training and linearizing a finite flow. While an extensive exploration is left for future work, initial tests indicate that the approach works similarly to the reverse Kullback-Leibler training.

[1] B. Joó, C. Jung, N. H. Christ, W. Detmold, R. Edwards, M. Savage, and P. Shanahan (USQCD Collaboration), Eur. Phys. J. A **55**, 199 (2019).
[2] M. S. Albergo, G. Kanwar, and P. E. Shanahan, Phys. Rev. D **100**, 034515 (2019).
[3] M. S. Albergo, D. Boyda, K. Cranmer, D. C. Hackett, G. Kanwar, S. Racanière, D. J. Rezende, F. Romero-López, P. E. Shanahan, and J. M. Urban, Phys. Rev. D **106**, 014514 (2022).
[4] R. Abbott, D. Boyda, D. C. Hackett, G. Kanwar, F. Romero-López, P. E. Shanahan, J. M. Urban, and M. S. Albergo, Proc. Sci., LATTICE2023 (**2024**) 011 [arXiv:2404.11674].
[5] C. Bonanno, A. Bulgarelli, E. Cellini, A. Nada, D. Panfalone, D. Vadacchino, and L. Verzichelli, J. High Energy Phys. 04 (2026) 051.
[6] S. Bacchio, Phys. Rev. D **108**, L091508 (2023).
[7] G. Catumba and A. Ramos, Eur. Phys. J. C **85**, 1037 (2025).
[8] R. Abbott, A. Botev, D. Boyda, D. C. Hackett, G. Kanwar, S. Racanière, D. J. Rezende, F. Romero-López, P. E. Shanahan, and J. M. Urban, Phys. Rev. D **109**, 094514 (2024).
[9] R. Abbott, D. Boyda, G. Kanwar, F. Romero-López, D. C. Hackett, P. E. Shanahan, and J. M. Urban, Proc. Sci., LATTICE2024 (**2025**) 066 [arXiv:2502.00263].
[10] R. Abbott, M. S. Albergo, A. Botev, D. Boyda, K. Cranmer, D. C. Hackett, G. Kanwar, A. G. D. G. Matthews, S. Racanière, A. Razavi, D. J. Rezende, F. Romero-López, P. E. Shanahan, and J. M. Urban, arXiv:2305.02402.
[11] M. S. Albergo and E. Vanden-Eijnden, arXiv:2410.02711.
[12] R. Horsley, R. Millo, Y. Nakamura, H. Perlt, D. Pleiter, P. E. L. Rakow, G. Schierholz, A. Schiller, F. Winter, and J. M. Zanotti (QCDSF and UKQCD Collaborations), Phys. Lett. B **714**, 312 (2012).
[13] G. Catumba, A. Ramos, and B. Zaldivar, Comput. Phys. Commun. **307**, 109396 (2025).
[14] D. J. Rezende and S. Mohamed, arXiv:1505.05770.
[15] L. Dinh, J. Sohl-Dickstein, and S. Bengio, arXiv:1605.08803.
[16] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, J. Mach. Learn. Res. **22**, 1 (2021).
[17] S. Kullback and R. A. Leibler, Ann. Math. Stat. **22**, 79 (1951).
[18] L. Vaitl, K. A. Nicoli, S. Nakajima, and P. Kessel, Mach. Learn. **3**, 045006 (2022).
[19] P. Bialas, P. Korcyl, and T. Stebel, Comput. Phys. Commun. **298**, 109094 (2024).
[20] M. Lüscher, Commun. Math. Phys. **293**, 899 (2010).
[21] Y. Nagai and A. Tomiya, Phys. Rev. D **111**, 074501 (2025).
[22] S. Bacchio, P. Kessel, S. Schaefer, and L. Vaitl, Phys. Rev. D **107**, L051504 (2023).
[23] M. Gerdes, P. de Haan, R. Bondesan, and M. C. N. Cheng, Phys. Rev. D **112**, 094516 (2025).
[24] Y. Nagai, H. Ohno, and A. Tomiya, Proc. Sci., LATTICE2024 (**2025**) 030 [arXiv:2501.16955].
[25] M. Favoni, A. Ipp, D. I. Müller, and D. Schuh, Phys. Rev. Lett. **128**, 032003 (2022).
[26] R. Abbott, M. S. Albergo, A. Botev, D. Boyda, K. Cranmer, D. C. Hackett, A. G. D. G. Matthews, S. Racanière, A. Razavi, D. J. Rezende, F. Romero-López, P. E. Shanahan, and J. M. Urban, Eur. Phys. J. A **59**, 257 (2023).
[27] G. Bhanot, U. M. Heller, and I. O. Stamatescu, Phys. Lett. B **129**, 440 (1983).
[28] R. Abbott, M. S. Albergo, D. Boyda, K. Cranmer, D. C. Hackett, G. Kanwar, S. Racanière, D. J. Rezende, F. Romero-López, P. E. Shanahan, B. Tian, and J. M. Urban, Phys. Rev. D **106**, 074506 (2022).
[29] T. A. DeGrand, Comput. Phys. Commun. **52**, 161 (1988).
[30] M. Luscher, in Les Houches Summer School: Session 93: Modern Perspectives in Lattice QCD: Quantum Field Theory and High Performance Computing (2010), pp. 331–399, arXiv:1002.4232.
[31] C. Urbach (European Twisted Mass Collaboration), Proc. Sci., LATTICE2007 (**2007**) 022 [arXiv:0710.1517].
[32] C. Bouchard, C. C. Chang, T. Kurth, K. Orginos, and A. Walker-Loud, Phys. Rev. D **96**, 014504 (2017).
[33] R. Abbott, D. C. Hackett, D. A. Pefkou, F. Romero-López, and P. E. Shanahan, Phys. Rev. Lett. **136**, 041901 (2026).
[34] A. Reuther, J. Kepner, C. Byun, S. Samsi, W. Arcand, D. Bestor, B. Bergeron, V. Gadepally, M. Houle, M. Hubbell *et al.*, 2018 IEEE High Performance extreme Computing Conference (HPEC) (2018), pp. 1–6, arXiv:1807.07814.

[35] A. Paszke *et al.*, arXiv:1912.01703.
[36] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, http://github.com/google/jax (2018).
[37] T. Hennigan, T. Cai, T. Norman, and I. Babuschkin, http://github.com/deepmind/dm-haiku (2020).
[38] A. Sergeev and M. Del Balso, arXiv:1802.05799.
[39] C. R. Harris, K. J. Millman, S. J. Van Der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith *et al.*, Nature (London) **585**, 357 (2020).
[40] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright *et al.*, Nat. Methods **17**, 261 (2020).
[41] J. D. Hunter, Comput. Sci. Eng. **9**, 90 (2007).