Covariant approximation averaging

Eigo Shintani,[1,2,*] Rudy Arthur,[3] Thomas Blum,[4,2] Taku Izubuchi,[5,2] Chulwoo Jung,[5] and Christoph Lehner[5]

[1]*PRISMA Cluster of Excellence, Institut für Kernphysik and Helmholtz Institute Mainz,*
Johannes Gutenberg-Universität Mainz, D-55099 Mainz, Germany
[2]*RIKEN-BNL Research Center, Brookhaven National Laboratory, Upton, New York 11973, USA*
[3]*CP³-Origins and the Danish Institute for Advanced Study DIAS, University of Southern Denmark,*
Campusvej 55, DK-5230 Odense M, Denmark
[4]*Physics Department, University of Connecticut, Storrs, Connecticut 06269-3046, USA*
[5]*High Energy Theory Group, Brookhaven National Laboratory, Upton, New York 11973, USA*

We present a new class of statistical error reduction techniques for Monte Carlo simulations. Using covariant symmetries, we show that correlation functions can be constructed from inexpensive approximations without introducing any systematic bias in the final result. We introduce a new class of covariant approximation averaging techniques, known as all-mode averaging (AMA), in which the approximation takes account of contributions of all eigenmodes through the inverse of the Dirac operator computed from the conjugate gradient method with a relaxed stopping condition. In this paper we compare the performance and computational cost of our new method with traditional methods using correlation functions and masses of the pion, nucleon, and vector meson in $N_f = 2 + 1$ lattice QCD using domain-wall fermions. This comparison indicates that AMA significantly reduces statistical errors in Monte Carlo calculations over conventional methods for the same cost.

I. INTRODUCTION

In order to increase the confidence we have in the results of a Monte Carlo simulation, a huge number of independent ensembles is always required. In lattice QCD many important observables suffer from notoriously large statistical errors due to fluctuations induced by the gauge fields used to compute expectation values, e.g., the neutron electric dipole moment [1–4], the hadronic contributions to the muon anomalous magnetic moment (g-2) [5], and the η-η' mass and mixing angle [6]. The precise determination of these observables, which provide important ingredients for the Standard Model (SM) and models beyond the SM, is a challenging task for lattice QCD. In this paper we present a detailed study of a new technique to efficiently evaluate correlation functions in a Monte Carlo simulation. An earlier publication by some of us already described the method and provided a few examples [7].

In lattice QCD, the numerical path integral is evaluated by Monte Carlo simulation to compute the expectation value of an observable $\mathcal{O}[U]$ given as the weighted average over configurations of gauge (gluon) fields, link variables U generated under probability distribution $P[U]$ on a lattice, in an ensemble,

$$\langle \mathcal{O} \rangle = \sum_U \mathcal{O}[U]P[U] = \sum_{i=1}^{N_{\text{conf}}} \frac{1}{N_{\text{conf}}} \mathcal{O}[U_i]$$
$$+ O(1/\sqrt{N_{\text{conf}}}) \quad \text{as } N_{\text{conf}} \to \infty. \quad (1)$$

To increase the accuracy of the ensemble average given the statistics of N_{conf} configurations, the development of numerical algorithms to efficiently compute observables is an important task. Traditionally translational symmetry of the correlation function is exploited to increase statistics,

$$\langle \mathcal{O}(x, y) \rangle = \langle \mathcal{O}(x^g, y^g) \rangle, \quad (2)$$

where the distance between operators on the shifted lattice sites is held constant, $\|x - y\| = \|x^g - y^g\|$. Ignoring statistical correlations between operators on shifted sites, the different N_G sets of $\mathcal{O}(x^g, y^g)$ with sink location x^g and source location y^g can be regarded as independent measurements. However, this naively requires N_G times the computational cost of a single measurement.

The original idea to avoid the cost of N_G measurements while still performing N_G shifts is low-mode averaging (LMA) [8–11], in which the inverse of the Dirac operator for each of $g \in G$ is computed from its low-lying eigenvectors. The benefit of LMA is that, once the low modes have been computed, the construction of the LMA estimator is not only low cost but also useful for low-mode deflation [12], i.e., as a preconditioner in the conjugate gradient (CG) method. There have been many lattice studies using LMA, primarily focused on low-mode dominated observables, for example low-energy constants in the ε-regime [13], or the chiral behavior of pseudoscalar mesons in the p-regime [14]. They have shown that there is some benefit from LMA for observables related to the pion. On the other hand, attempts to use LMA for baryons

*shintani@kph.uni-mainz.de

or heavy mesons [15–17] were not as successful, presumably because these states are not dominated by a relatively small number of low modes (we also found recent attempts to use an extended method called low-mode substitution for baryon spectroscopy in [18]).

Recently we extended the LMA idea to efficiently handle the vast majority of hadronic states that are not dominated by low modes [7]. The idea is to include all modes of the Dirac operator but with dramatically reduced computational cost compared to the usual CG method. By using covariant symmetries, approximate (and therefore inexpensive) correlation functions are used to compute expectation values without introducing any systematic error (bias). *All-mode-averaging* (AMA) in which a relaxed stopping condition of the CG is employed as in [19] takes the contributions of all modes into account. The method is broadly applicable to other fields using Monte Carlo simulation, e.g., many-body systems, atomic systems, and cold gas systems (see [20–24]). This paper gives a detailed description of the covariant approximation averaging (CAA) with primary examples, LMA and AMA [7]. We also present several numerical results with high precision and cost-performance comparison with standard methods.

This paper is organized as follows: in the next section we describe the CAA procedure and compare LMA and AMA. In Sec. III we show numerical results for AMA using domain-wall fermions and compare to LMA and the standard multisource method. In Sec. IV we present several examples extending the approximation and the results of some numerical tests. In the last section we summarize and discuss further extensions of AMA. In Appendix B, the possible small biases of AMA due to finite precision floating point arithmetic are discussed, and we present how to remove them completely in Appendix C.

II. COVARIANT APPROXIMATION AVERAGING

A. General argument

Under a symmetry transformation $g \in G$, the expectation value of the transformed functional $\mathcal{O}[U]$ (for example, a hadron propagator) is equivalent to that computed on the transformed configuration U^g

$$\langle \mathcal{O}^g[U] \rangle = \langle \mathcal{O}[U^g] \rangle, \tag{3}$$

where $U^g(x) = U(x^g)$, while translational symmetry of the observable is expressed as $\mathcal{O}^g[U](x, y) = \mathcal{O}[U](x^g, y^g)$. If $\mathcal{O}[U]$ is covariant under the symmetry, on each gauge configuration

$$\mathcal{O}^g[U] = \mathcal{O}[U^g], \tag{4}$$

then there is the trivial identity

$$\sum_{g \in G} \mathcal{O}^g[U] = \sum_{g \in G} \mathcal{O}[U^g], \tag{5}$$

for a set of transformations $g \in G$ whose number of elements is N_G. From Eqs. (3), (4), and (5), an average over a set of symmetry transformations is defined as

$$\mathcal{O}_G[U] \equiv \frac{1}{N_G} \sum_{g \in G} \mathcal{O}^g[U] = \frac{1}{N_G} \sum_{g \in G} \mathcal{O}[U^g], \tag{6}$$

and one sees that $\langle \mathcal{O}_G[U] \rangle$ is identical to $\langle O[U] \rangle$, since any transformed configuration U^g appears with the same probability as U in the Monte Carlo simulation with an action invariant with respect to g. Note the statistical error of $\mathcal{O}_G$ decreases by a factor $1/\sqrt{N_G}$ times smaller, while its computational cost increases by a factor N_G times more.

In order to reduce the computational cost implied by Eq. (6), we introduce an approximation for $\mathcal{O}$, which is called $\mathcal{O}^{(\text{appx})}$. Averaging over $g \in G$ as in Eq. (6) for $\mathcal{O}^{(\text{appx})}$ yields

$$\mathcal{O}_G^{(\text{appx})} = \frac{1}{N_G} \sum_{g \in G} \mathcal{O}^{(\text{appx})g}. \tag{7}$$

Using $\mathcal{O}^{(\text{appx})}$ and the original $\mathcal{O}$, an improved estimator for $\mathcal{O}$ is defined by

$$\mathcal{O}^{(\text{imp})} = \mathcal{O} - \mathcal{O}^{(\text{appx})} + \mathcal{O}_G^{(\text{appx})}$$
$$\equiv \mathcal{O}^{(\text{rest})} + \mathcal{O}_G^{(\text{appx})}, \tag{8}$$

$$\mathcal{O}^{(\text{rest})} = \mathcal{O} - \mathcal{O}^{(\text{appx})}. \tag{9}$$

(In the definition of $\mathcal{O}^{(\text{rest})}$, we used the unit element of G; however, any other elements would serve the purpose just as well.) Since $\mathcal{O}^{(\text{appx})}$ in $\mathcal{O}^{(\text{imp})}$ is canceled by $\mathcal{O}_G^{(\text{appx})}$ after performing the path integral and using the covariance of $\mathcal{O}^{(\text{appx})}$ as in Eq. (4), one easily sees that the expectation value of the improved estimator agrees with the original,

$$\langle \mathcal{O}^{(\text{imp})} \rangle = \langle \mathcal{O} \rangle. \tag{10}$$

As shown in Appendix A, using the standard deviations of $\mathcal{O}$, σ, the approximation $\mathcal{O}^{(\text{appx})}$, $\sigma^{(\text{appx})}$, and the transformed approximation $\mathcal{O}^{(\text{appx})g}$, $\sigma^{(\text{appx})g}$, where $\sigma^X = \sqrt{\langle (\Delta \mathcal{O}^X)^2 \rangle}$, and $\Delta \mathcal{O}^X = \mathcal{O}^X - \langle O^X \rangle$, and the correlations are defined by

$$r_g = \frac{\langle \Delta \mathcal{O} \Delta \mathcal{O}^{(\text{appx})g} \rangle}{\sigma \sigma_g^{(\text{appx})}}, \tag{11}$$

$$r_{gg'}^{\text{corr}} = \frac{\langle \Delta \mathcal{O}^{(\text{appx})g} \Delta \mathcal{O}^{(\text{appx})g'} \rangle}{\sigma^{(\text{appx})g} \sigma^{(\text{appx})g'}}, \tag{12}$$

the standard deviation of the improved estimator is

$$\sigma^{(\text{imp})} \simeq \sigma \left[2\Delta r + \frac{1}{N_G} - \frac{2}{N_G}\Delta r + R^{\text{corr}} \right]^{1/2}, \quad (13)$$

$$R^{\text{corr}} = \frac{1}{N_G^2} \sum_{g \neq g'} r_{gg'}^{\text{corr}}, \quad (14)$$

with $\Delta r = 1 - r$, $r \equiv r_{g=I}$. Note that, in Eq. (13), we approximate $\sigma \simeq \sigma^{(\text{appx})}$, and the correlation between $\mathcal{O}$ and $\mathcal{O}^{(\text{appx})g}$ is similar to that for $\mathcal{O}^{(\text{appx})}$, i.e., $r_{g \neq I}^{\text{corr}} \simeq r_{g \neq I}$ (which assumes that there is a strong correlation between $\mathcal{O}$ and $\mathcal{O}^{(\text{appx})}$). In [7,25], we also ignored the third and fourth terms in (13). In the equation above, Δr and r_g^{corr} indicate the quality of the approximation and the magnitude of the correlation among the $\{\mathcal{O}^{(\text{appx})g}\}_{g \in G}$, respectively. To achieve a reduction of the statistical error of magnitude $\sim 1/\sqrt{N_G}$, an $\mathcal{O}^{(\text{appx})}$ with small Δr and small positive $r_{gg'}^{\text{corr}}$ is necessary. Furthermore, the cost of computing $\mathcal{O}^{(\text{appx})}$ should be much cheaper than $\mathcal{O}$.

Taking the consideration above into account, we impose the following conditions on $\mathcal{O}^{(\text{appx})}$ and the choice of transformation, $g \in G$, for $\mathcal{O}^{(\text{appx})g}$:

CAA-1: $\mathcal{O}^{(\text{appx})}$ is covariant under G as in Eq. (4).[1]

CAA-2: $\mathcal{O}^{(\text{appx})}$ is strongly correlated with $\mathcal{O}$, i.e., $\Delta r \ll 1$.

CAA-3: The computational cost of $\mathcal{O}^{(\text{appx})}$ is much smaller than $\mathcal{O}$.

CAA-4: The transformation $g \in G$ is chosen to give small (compared to $1/N_G$) positive correlations among $\{\mathcal{O}^{(\text{appx})g}\}_{g \in G}$, i.e., $R^{\text{corr}} \ll 1/N_G$.

Note that the last condition is not necessary if the cost of constructing $\mathcal{O}^{(\text{appx})}$ is negligible (so that, in [7], we have not included the last condition). The tuning of the most appropriate $\mathcal{O}^{(\text{appx})}$ for the target observable is important to maximize the reduction of the statistical error. In the following, we show two examples of CAA in lattice QCD.

B. Example: Low-mode averaging

In lattice QCD, $\mathcal{O}$ is a hadron correlator, given as the product of inverses of the Dirac operator ($S[U]$). In LMA, the approximation defined as $\mathcal{O}^{(\text{appx})} = \mathcal{O}^{(\text{LMA})}$ is constructed by

$$\mathcal{O}^{(\text{LMA})} = \mathcal{O}[S^{(\text{low})}], \quad \mathcal{O}_G^{(\text{LMA})} = \frac{1}{N_G} \sum_{g \in G} \mathcal{O}[S^{(\text{low})g}], \quad (15)$$

$$S^{(\text{low})}(x,y) = \sum_{k=1}^{N_\lambda} \lambda_k^{-1} \psi_k(x) \psi_k^\dagger(y), \quad (16)$$

---

[1]As explained in Appendix B, this condition is not necessary to fulfill Eq. (3) if we introduce a randomly chosen shift of source location in Appendix C.

with low-lying eigenmodes ψ_k and eigenvalues λ_k of the Hermitian Dirac matrix $H(x,y)$, $\sum_y H(x,y)\psi_k(y) = \lambda_k \psi_k(x)$. For low-mode dominant observables, like the pion propagator and related form factors, the eigenmodes with small $|\lambda_k|$ saturate the observable, and thus r in Eq. (11) may be close to unity (*CAA-2*). $\mathcal{O}^{(\text{LMA})}$ is covariant since $H[U^g](x,y) = H[U](x^g, y^g)$; we have $\mathcal{O}^g[S^{(\text{low})}[U]] = \mathcal{O}[S^{(\text{low})}[U^g]]$ (*CAA-1*). The construction of $\mathcal{O}_G^{(\text{appx})}$ requires an inner product of the low-mode and source (sink) vectors and a complex times vector multiply. Since the construction of $\mathcal{O}_G^{(\text{appx})}$ is cheap, the statistical error of low-mode dominant observables is significantly reduced (*CAA-3*) [10,11] [because the computational cost of $O^{g,(\text{LMA})}$ is small, condition (*CAA-4*) is not so important].

C. Example: All-mode averaging

AMA is similarly defined as

$$\mathcal{O}^{(\text{AMA})} = \mathcal{O}[S^{(\text{all})}], \quad \mathcal{O}_G^{(\text{AMA})} = \frac{1}{N_G} \sum_{g \in G} \mathcal{O}[S^{(\text{all})g}], \quad (17)$$

$$S^{(\text{all})}b = \sum_{k=1}^{N_\lambda} \lambda_k^{-1}(\psi_k^\dagger b)\psi_k + f_\varepsilon(H)b, \quad (18)$$

$$f_\varepsilon(H)b = \sum_{i=1}^{N_{\text{CG}}} (H)^i c_i, \quad (19)$$

where $f_\varepsilon b$ is a polynomial of H with vector "coefficients" c_i. In practice this combination is obtained from the CG, depending on the source vector b and initial guess x_0. The subscript ε indicates the norm of the residual vector after N_{CG} iterations, or steps, of the CG.

In AMA, the (exact) low-mode contribution to the propagator within the range $[\lambda_1, \lambda_{N_\lambda}]$ is taken into account by projecting the source vector b onto the orthogonal subspace,

$$b_{\text{proj}} \equiv \left(1 - \sum_{k=1}^{N_\lambda} \psi_k \psi_k^\dagger \right)b, \quad (20)$$

where the low mode is normalized as $\sum_x \psi_k^\dagger(x)\psi_k(x) = 1$. By adopting the above projected source vector into the CG process (see Algorithm 1), we obtain the solution x_{CG},

$$x_{\text{CG}} + \sum_{k=1}^{N_\lambda} \lambda_k^{-1}(\psi_k^\dagger b)\psi_k = S^{(\text{all})}b. \quad (21)$$

Notice that the CG is deflated at the same time. Further, the higher mode contribution $(\lambda_{N_\lambda} < \lambda \leq \lambda_{\text{max}})$ is treated approximately, $f_\varepsilon(\lambda) \approx 1/\lambda$, by using the relaxed stopping criterion in the CG. Therefore the computational cost of $f_\varepsilon(H)$ is significantly smaller than the usual CG used in $\mathcal{O}$

(*CAA-3*). Compared to LMA, in which eigenmodes with $\lambda > \lambda_{N_\lambda}$ are ignored, AMA introduces f_ε to take into account the contribution of all higher modes, and thus the quality of the approximation to $\mathcal{O}$ is greatly improved (*CAA-2*). In Eq. (17) the covariance (*CAA-1*) is also fulfilled since $f_\varepsilon(H)$ is covariant under the transformation g; $f_\varepsilon^g(H(x, y)) = f_\varepsilon(H(x^g, y^g))$.

Here we consider two choices of the stopping condition in the CG:

   (i) the norm of the residual vector is smaller than some prescribed value,

   (ii) there is a fixed number of CG iterations.

The first condition naturally controls the accuracy of the CG and thus the approximation $\mathcal{O}^{(\mathrm{appx})}$, and in this paper we have employed it as the stopping condition. However, it may happen that this criterion introduces a violation of covariant symmetry as systematic bias due to numerical round-off error, for example, because of the order of operations in one's code.[2] As described in detail in Appendix B, this bias is orders of magnitude smaller than the statistical error in practice. In the same appendix, we also present an argument to reduce the bias by fixing the number of CG iterations instead of fixing the CG stopping condition for the residual vector norm. Note that f_ε can also be computed directly from a polynomial with fixed coefficients rather than dynamically computed in the CG.

We emphasize, as in [7] and demonstrated in Appendix B 2, when using AMA it is mandatory to compute the size of the violation of covariance on a small number of configurations to ensure that the bias is negligible. Alternatively, one can *completely* remove the bias by using randomly selected source locations as described in Appendix C.

Figure 1 illustrates the spectral decomposition of $\mathcal{O}^{(\mathrm{LMA})}$ defined in Eq. (16) and $\mathcal{O}^{(\mathrm{AMA})}$ defined in Eq. (18). In AMA, because we use the exact low-lying eigenvectors, the behavior in the low-mode region is consistent with LMA. The number of intersections with the exact solution corresponds to the polynomial degree in the approximation which is equal to the number of CG iterations. The discrepancy with the exact solution can be controlled by the number of low modes used in deflation and the degree of the polynomial [see Eq. (18)].

The correlation among $\mathcal{O}^g$ will not be significant if we choose appropriate transformations, $g \in G$, for instance, by widely separating source points among $\{\mathcal{O}^g\}_{g \in G}$, so that the $r_{gg'}^{\mathrm{corr}}$ term in Eq. (13) is negligible (*CAA-4*). Unlike LMA, AMA entails non-negligible additional cost to construct $S^{(\mathrm{all})}$ (fourth step of the AMA algorithm in Table I), and hence the judicious tuning of N_G and choice of $g \in G$ are important to reduce the computational cost.

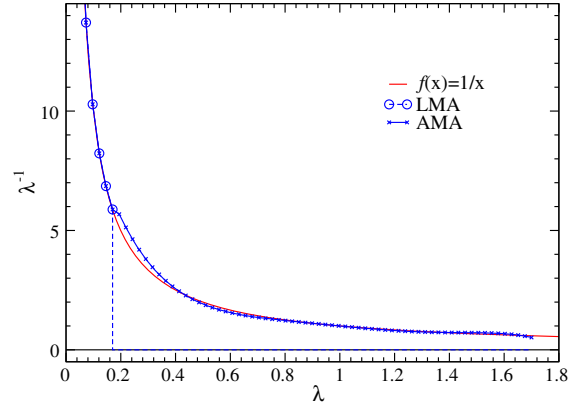[2]We thank both M. Lüscher and S. Hashimoto who, independently, pointed this out.

FIG. 1 (color online). Approximations for the spectral decomposition of the quark propagator in LMA (circle-dashed line) and AMA (cross-solid line). The x axis denotes the eigenvalue of the Hermitian Dirac matrix. The circle symbol corresponds to $\mathcal{O}^{(\mathrm{LMA})}$ and the blue solid line corresponds to $\mathcal{O}^{(\mathrm{AMA})}$. The red solid line shows the exact solution.

III. NUMERICAL RESULTS

In this section we show the numerical comparison between the standard method and AMA/LMA for the hadron spectrum and the form factors of the nucleon using realistic lattice QCD parameters.

A. Setup

We use the $N_f = 2 + 1$ domain-wall fermion (DWF) configurations generated by the RBC/UKQCD Collaboration on a $24^3 \times 64$ lattice, with gauge coupling $\beta = 2.13$ for the Iwasaki gauge action [26]. The CG algorithm with four-dimensional even-odd preconditioning (see Appendix E) was used to compute quark propagators at quark mass $m = 0.005$ and 0.01, corresponding to 0.33 and 0.42 GeV pion masses, respectively, and the fifth dimension size for DWF is $L_s = 16$.

To calculate the eigenvectors of the Hermitian even-odd preconditioned DWF operator, we implement the implicitly restarted Lanczos algorithm with Chebychev polynomial acceleration [27–30]. In Appendix D we describe the detailed implementation. The degree of the Chebychev polynomial in the Lanczos method is 100, and the parameters $(\alpha, \beta) = (0.04, 1.68)$ for $m = 0.005$ and $(\alpha, \beta) = (0.025, 1.68)$ for $m = 0.01$ are chosen to rapidly converge the "wanted" part of the spectrum, here the lowest few hundred modes [see Eqs. (D9) and (D11)]. In the implicitly restarted Lanczos method, we label N_λ the number of wanted eigenvectors and $p = 40$ the number of unwanted vectors (see Appendix D). We compute the exact low modes of Hermitian four-dimensional (4D) even-odd preconditioned DWF Dirac operator, H_{4Deo}, to better than 10^{-10} numerical accuracy, $\|(H_{\mathrm{4Deo}} - \lambda_k)\psi_k\|/\|\psi_k\| < 10^{-10}$. In Table II we summarize the parameters in the Lanczos method, the number of

TABLE I. LMA and AMA algorithms.

LMA algorithm	AMA algorithm
1: Compute low modes ψ_k of H	1: If $\lambda_{N_\lambda} \neq 0, N_\lambda > 0$ Compute low modes ψ_k of H
2: Set source vector b and G-invariant initial guess x_0	
3: Compute accurate x_{CG} and $\mathcal{O}[S]$ precisely [use deflation method in Eqs. (20) and (21) if ψ_k exits]	
4: Compute $S^{(\text{low})}b$ in (16) and $\mathcal{O}^{(\text{LMA})} = \mathcal{O}[S^{(\text{low})}]$	4: Compute $S^{(\text{all})}b$ in (18) and $\mathcal{O}^{(\text{AMA})} = \mathcal{O}[S^{(\text{low})}]$ using deflated CG (if $\lambda_{N_\lambda} \neq 0$)
5: $\mathcal{O}^{(\text{rest})} = \mathcal{O}[S] - \mathcal{O}[S^{(\text{low})}]$	5: $\mathcal{O}^{(\text{rest})} = \mathcal{O}[S] - \mathcal{O}[S^{(\text{all})}]$;
6: Set shifted source b^g and G-invariant initial guess x_0^g	
7: Average $\mathcal{O}_G^{(\text{LMA})} = \mathcal{O}[S^{(\text{low})}]$ over $g \in G$ to get $\mathcal{O}_G^{(\text{LMA})}$	7: Average $\mathcal{O}_G^{(\text{AMA})} = \mathcal{O}[S^{(\text{all})}]$ over $g \in G$ to get $\mathcal{O}_G^{(\text{AMA})}$
8: $\mathcal{O}^{(\text{imp})} = \mathcal{O}^{(\text{rest})} + \mathcal{O}_G^{(\text{appx})}$	

gauge configurations N_{conf} in each ensemble, and the number of low modes N_λ computed on each configuration.

In AMA/LMA, the set of transformations $g \in G$ in Eq. (7) are taken as translational symmetry. The estimator $\mathcal{O}_G^{(\text{appx})}$ is obtained with $N_G = 32$ different source locations, separated by 12 sites for spatial directions and 16 sites for the temporal direction, starting from the origin, i.e., at positions (0, 0, 0, 0), (12, 0, 0, 0), (12, 12, 0, 0), …, (12, 12, 12, 48) in lattice units. This setup is used for measurements on configurations separated by 40 Hybrid Monte Carlo trajectories. In addition, measurements are made on a second set of configurations, also separated by 40 trajectories, but lying in between configurations of the first set. On the second set, all source locations are shifted by the lattice vector (6, 6, 6, 0) with respect to the original functional $\mathcal{O}$. In the CG, the norm of the residual vector is defined as $\|H_{\text{4Deo}}x - b\|/\|b\|$ with source vector b and solution vector x_{CG} (see also Table II). For the stopping conditions for the exact CG and the relaxed CG we have $\varepsilon = 10^{-8}$ and $\varepsilon = 0.003$, respectively.[3]

We use gauge-invariant Gaussian smeared sources with the same parameters as in Ref. [31] to compare the performance of LMA and AMA. In [31], the authors measured three- and two-point functions for four source locations in the temporal direction to extract the nucleon isovector form factors and axial charge, and thus $4 \times N_{\text{conf}}$ samples were accumulated. For $m = 0.005$, quark sources set on two time slices separated by 32 sites were used (double source method) to efficiently double the statistics. Reference [31] also employed nonrelativistic nucleon sources (2 quark spins rather than 4) to reduce the computational cost further, while in our case we use relativistic sources. Therefore, in the analysis below, we account for these two factors to ensure a fair comparison of statistical errors.

B. Computational cost estimate

In order to compare the computational cost between the standard method and LMA/AMA, we use the number of applications of H_{4Deo}^2 (#Mult in Table III) to estimate total costs in each case. In the standard method, the cost without deflation is $\#\text{Mult}_{\text{CG(org)}}$ times the number of color and spin sources used per configuration,

$$\text{Cost(org)} = \#\text{Mult}_{\text{CG(org)}} \times 12 \times N_{\text{conf}}. \quad (22)$$

On the other hand, when deflating the Dirac operator, the cost is

$$\begin{aligned}\text{Cost}_{\text{w/defl}}(\text{org}) = (\#\text{Mult}_{\text{Lanczos}} + \#\text{Mult}_{\text{deflCG(org)}} \times 12) \\ \times N_{\text{conf}}, \quad (23)\end{aligned}$$

where we add the cost of the Lanczos process to obtain the low modes. We note that, based on wall-clock timing, the time for multiplication of the Dirac operator dominates the Lanczos step, and Gram-Schmidt reorthogonalization is negligible due to the O(100) degree of the Dirac matrix polynomial. Therefore, we use the number of multiplications of the polynomial of the Dirac operator as a good representative of the computational cost.

In LMA, ignoring the small cost of constructing the approximation $\mathcal{O}^{(\text{LMA})}$ and $\mathcal{O}_G^{(\text{LMA})}$ from the low modes, the total cost is the same as $\text{Cost}_{\text{w/defl}}(\text{org})$,

TABLE II. Parameters of LMA/AMA in each ensemble. (α, β) is the input range of Chebychev polynomial in the Lanczos method with N_λ wanted and 40 unwanted eigenmodes. We present the absolute value of the minimum eigenvalue as $|\lambda_1|$ and N_λth eigenvalue $|\lambda_{N_\lambda}|$ up to the first significant figure in each ensemble. "#Restart" column shows the range of numbers of restarted Lanczos iterations.

m	N_{conf}	N_λ	(α, β)	$|\lambda_1|$	$|\lambda_{N_\lambda}|$	#Restart
0.005	398	400	(0.04, 1.68)	0.004	0.04	5–6
0.01	348	180	(0.025, 1.68)	0.006	0.02	5–6

---

[3]Note that when using an even-odd basis, one needs to choose the four-dimensional shift vector of the source point to avoid breaking *CAA-3*. Shifts that end on an even (odd) point for even (odd) sites are sufficient.

TABLE III.   The number of multiplications of kernel $H_{4\mathrm{Deo}}^2$. "$\#\mathrm{Mult}_{\mathrm{Lanczos}}$" is its number in 5 restarting Lanczos processes. We also show the range of #Mult with and without the deflation method for exact calculation ($\#\mathrm{Mult}_{\mathrm{deflCG(org)}}$, $\#\mathrm{Mult}_{\mathrm{CG(org)}}$) and approximation in AMA ($\#\mathrm{Mult}_{\mathrm{deflCG(AMA)}}$) using the low mode of $H_{4\mathrm{Deo}}^2$.

m	$\#\mathrm{Mult}_{\mathrm{Lanczos}}$	$\#\mathrm{Mult}_{\mathrm{CG(org)}}$	$\#\mathrm{Mult}_{\mathrm{deflCG(org)}}$	$\#\mathrm{Mult}_{\mathrm{deflCG(AMA)}}$
0.005	64K	3K	350–360	70–90
0.01	42K	2K	600–630	90–130

$$\mathrm{Cost(LMA)} = \mathrm{Cost}_{\mathrm{w/defl}}(\mathrm{org}). \qquad (24)$$

In AMA, there are three parts to the total cost, the eigenvector computation, the exact CG solve, and N_G relaxed CG solves, so the total cost reads

$$
\begin{aligned}
\mathrm{Cost(AMA)} \\
= (\#\mathrm{Mult}_{\mathrm{Lanczos}} + (\#\mathrm{Mult}_{\mathrm{deflCG(org)}} + \#\mathrm{Mult}_{\mathrm{deflCG(AMA)}} \\
\times N_G) \times 12) \times N_{\mathrm{conf}}. \qquad (25)
\end{aligned}
$$

In the following section, to compare costs of LMA/AMA to the standard method, we define the cost ratio multiplied with the squares of statistical error ratio to obtain a normalized cost, i.e., one that reflects the cost to achieve the same error,

$$r_{\mathrm{Cost}}^{\mathrm{w/odefl}} = \frac{\mathrm{Cost(LMA/AMA)}}{\mathrm{Cost(org)}} r_{\mathrm{Error}}^2, \qquad (26)$$

$$r_{\mathrm{Cost}}^{\mathrm{w/defl}} = \frac{\mathrm{Cost(LMA/AMA)}}{\mathrm{Cost}_{\mathrm{w/defl}}(\mathrm{org})} r_{\mathrm{Error}}^2, \qquad (27)$$

$$r_{\mathrm{Error}} = \frac{\mathrm{Error(LMA/AMA)}}{\mathrm{Error(org)}}. \qquad (28)$$

C. Hadron spectrum

First we show results for hadron propagators obtained by using the standard method and LMA/AMA with parameters given in the previous section. Figure 2 shows that the error reduction achieved with AMA is close to the ideal rate, $1/\sqrt{N_G} \simeq 0.18$ for nucleon, pion, and vector propagators, for source-sink separations $t = 4$, 8, and 12 (nucleon and vector), and $t = 4$, 20, and 25 (pion), while LMA does not work well at short distance ($t = 4$) except for the pion. Since low modes dominate the pion propagator, LMA and AMA show similar error reduction. For AMA we see that $\mathcal{O}^{(\mathrm{imp})}$ is close in value to $\mathcal{O}_G^{(\mathrm{appx})}$, while in LMA the difference is much larger, especially for short distances (except for the pion propagator). It turns out that AMA provides a good approximation to the original and clearly shows that AMA can reduce statistical errors for both long and short distances by approximating the quark propagator with $f_\varepsilon(H)$ obtained with the relaxed CG for the high part of the Dirac spectrum.

In Fig. 3, we plot $r_{gg'}^{\mathrm{corr}}$ against the distance between source locations on a given time slice and R^{corr} for zero momentum nucleon, pion, and vector meson propagators. These quantities are important for choosing N_G and the
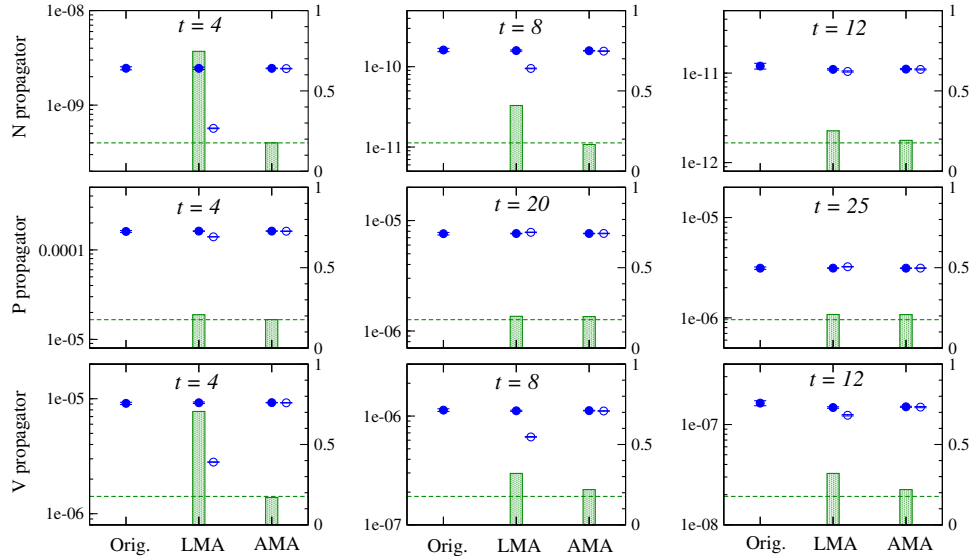

FIG. 2 (color online).   The propagator of nucleon (top), pion (middle), and vector meson (bottom) at time separation $t = 4, 8, 12$ for nucleon and vector meson, and $t = 4, 20, 25$ for pion. We show the values of these propagators used in original, LMA, and AMA. The filled symbols are results of the improved estimator $\mathcal{O}^{(\mathrm{imp})}$ and open symbols are results of the averaged approximation $\mathcal{O}_G^{(\mathrm{appx})}$. The bar in AMA/LMA shows the ratio of the relative error with the original one. This value corresponds to the right-perpendicular axis. The horizontal bar shows the ideal ratio of relative error $1/\sqrt{32} \simeq 0.18$ in the case of no correlation between spatial source locations.
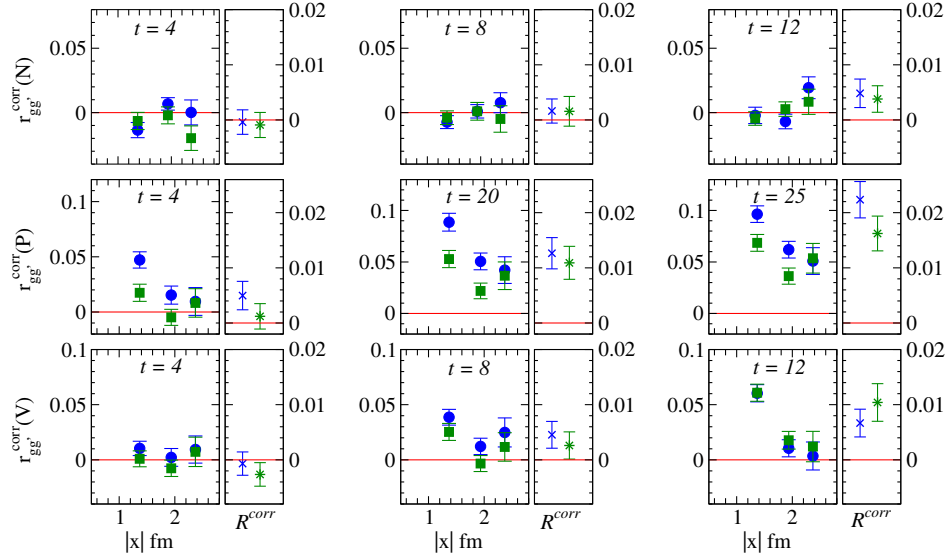
FIG. 3 (color online).   The correlation $r_{gg'}^{corr}$ and R^{corr} as a function of physical spatial distance for source locations between $\mathcal{O}^{(appx)g}$ and $\mathcal{O}^{(appx)g'}$. We take the average over $r_{gg'}^{corr}$ in the same spatial distance with identical temporal source locations. The top panel is for the propagator of the nucleon, the middle is for the pion, and the bottom is of the vector at time slice $t = 4, 8, 12$ for nucleon and vector meson $t = 4, 20, 25$ for pion. The blue (green) symbols are in $m = 0.005$ (0.01).
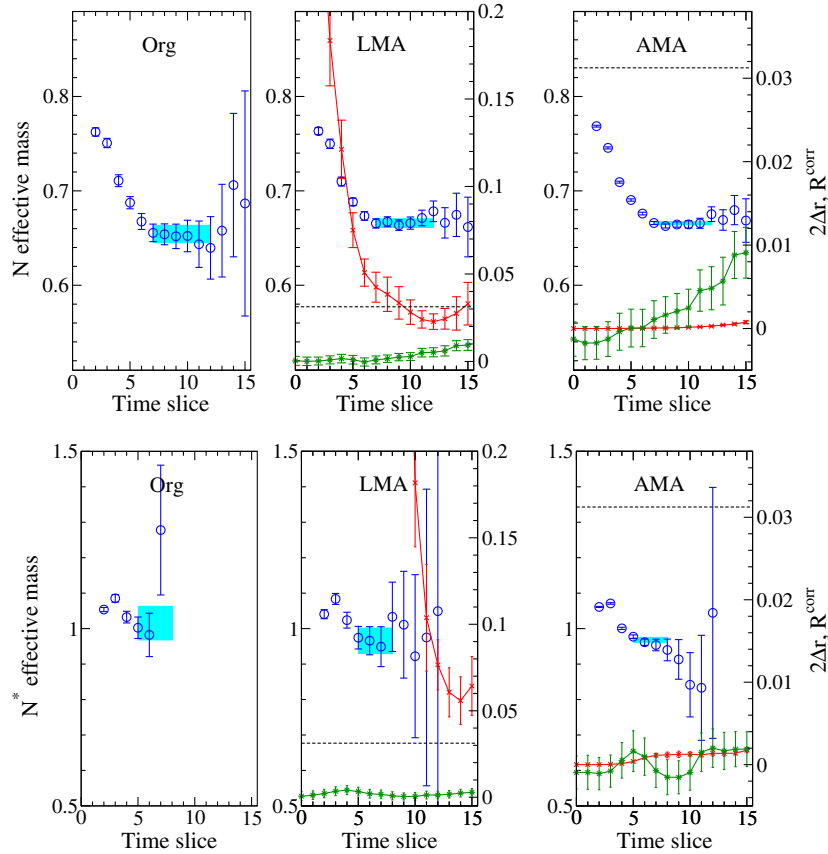
FIG. 4 (color online).   The effective mass plot of nucleon propagator (top) and its parity partner (bottom) with smeared source and point sink using original (left panel), LMA (middle panel), and AMA (right panel) at $m = 0.005$. The cross symbols show the magnitude of $2\Delta r$, and the star symbols denote R^{corr}, as defined in Eq. (11). The right-perpendicular axis corresponds to this value. The dashed line shows the value of $1/N_G$.

transformations $g \in G$ to efficiently implement CAA as explained in Sec. II. One sees that at the smallest separation from the origin [in which the source location is (12, 0, 0, 0), (0, 12, 0, 0), and (0, 0, 12, 0)] there is significant correlation compared to the case of large separation. This behavior becomes apparent when the hadron propagates far away from the source location (large t). Comparing the different masses, especially for the pion propagator, $r_{gg'}^{\mathrm{corr}}$ is larger for lighter mass. For the nucleon and vector meson propagators R^{corr}, which is the sum of $r_{gg'}^{\mathrm{corr}}$ divided by N_G^2, is relatively small compared to $1/N_G \simeq 0.031$ in Eq. (13), and therefore in our setting of $g \in G$ the reduction of statistical error is close to the ideal ratio, $1/\sqrt{N_G} \simeq 0.18$. We notice that for the pion propagator R^{corr} is relatively large since $r_{gg'}^{\mathrm{corr}}$ increases when the pion propagates a large distance. More details will be discussed below.

In Figs. 4 and 5, we plot the effective mass of several hadron channels together with $2\Delta r$ and R^{corr} defined in Eqs. (11) and (14). As previously discussed, an approximation having strong correlation with $\mathcal{O}$ has small $2\Delta r$. In the case of AMA the effective mass for both the nucleon and vector mesons is improved over LMA, especially for t less than 15 where $2\Delta r$ is less than 0.1%. On the other hand, R^{corr} of AMA within the fitting region is similar to R^{corr} of LMA, and it is less than 20% of $1/N_G$ for the

nucleon (and its parity partner N^*, which is given by the negative parity projection for the nucleon two point function) and the vector meson. (More detailed discussion and recent lattice study refers to, for example, [32,33] and references therein.) Thus the two contributions in Eq. (13), $2\Delta r$ and R^{corr}, are negligible compared to $1/\sqrt{N_G}$, and therefore the error reduction of these hadron masses is close to $1/\sqrt{N_G}$ in AMA (see Table IV). However, for the pion propagator, we observe that $2\Delta r$ in AMA at below $t = 5$ is much smaller than LMA; otherwise at $t > 5$ both cases become similarly tiny as seen in Fig. 5. On the other hand, R^{corr} of the pion propagator is similar between LMA and AMA, with magnitude around 40%–90% of $1/N_G$. As the consequence the error reduction of AMA for pion propagator and pion mass is similar in magnitude with LMA in a region where the pion ground state dominates. We note that the relatively large correlation between different source locations for the pion propagator may result in a slightly smaller error reduction of the pion mass (see the "m_π" row in Table IV).

In Tables IV and V we compare the fit results of hadron masses and scaled costs of LMA/AMA to achieve the same statistical error of the standard method. Here we use the chi-squared fitting with a single exponential function including the correlation in the temporal direction. χ^2/dof
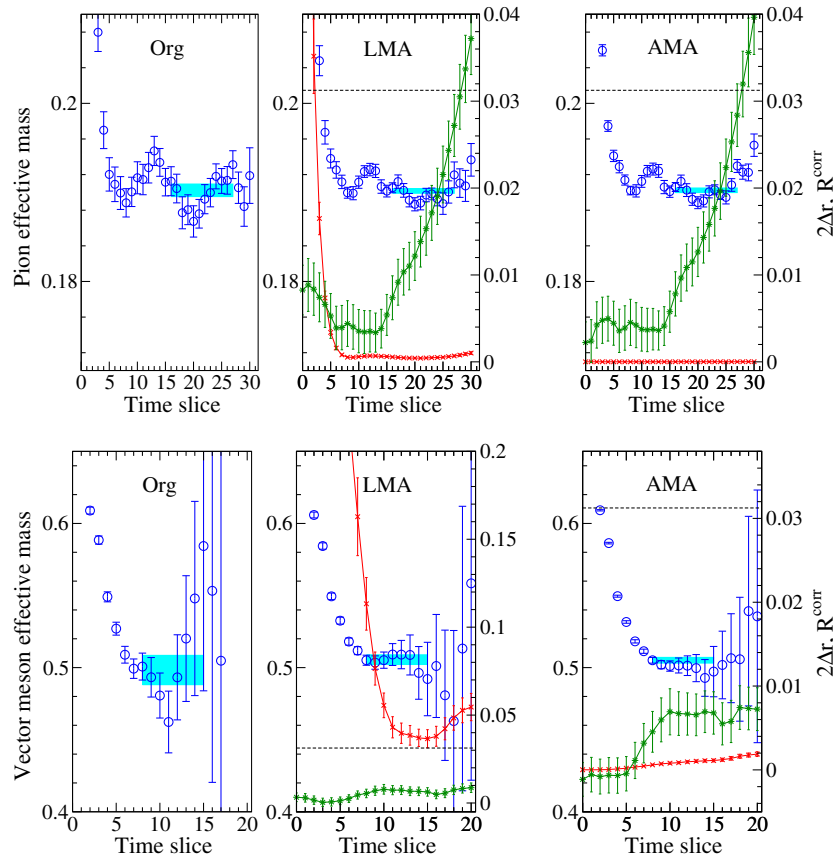

FIG. 5 (color online).   Similar plot to Fig. 4 of the effective mass of pion (top) and vector meson (bottom).

TABLE IV. The comparison of hadron mass (nucleon with momenta, pion, vector meson, and parity partner of nucleon) in GeV units obtained by the global fit of the correlator (point sink and gauge-invariant Gaussian smeared source) in the AMA/LMA method with $N_G = 32$. For reference we show the result with the correlator in a single source location. "Cost" column shows the ratio of the computational cost of AMA/LMA and the original one after scaling to the same accuracy. We also compare the cost with and without the deflation method in the original calculation using the number of low mode presented in Table II.

$m = 0.005$									
	Org	LMA	r_{Error}	$r_{\text{Cost}}^{\text{w/o defl}}$	$r_{\text{Cost}}^{\text{w/defl}}$	AMA	r_{Error}	$r_{\text{Cost}}^{\text{w/o defl}}$	$r_{\text{Cost}}^{\text{w/defl}}$
Fit: [7, 12]									
m_N	1.1322(156)	1.1520(78)	0.50	0.48	0.25	1.1519(27)	0.17	0.08	0.04
$E_N(n_p^2 = 1)$	1.2072(172)	1.2349(82)	0.48	0.43	0.23	1.2393(30)	0.18	0.09	0.04
$E_N(n_p^2 = 2)$	1.3095(232)	1.3171(96)	0.42	0.33	0.17	1.3229(39)	0.17	0.08	0.04
$E_N(n_p^2 = 3)$	1.3723(436)	1.3941(135)	0.31	0.18	0.10	1.4010(55)	0.13	0.05	0.02
$E_N(n_p^2 = 4)$	1.5205(627)	1.4638(192)	0.31	0.18	0.09	1.4726(88)	0.14	0.05	0.03
Fit: [5, 8]									
m_{N^*}	1.757(81)	1.671(61)	0.75	1.07	0.56	1.675(11)	0.15	0.06	0.03
Fit: [16, 27]									
m_π	0.3291(12)	0.3290(4)	0.37	0.27	0.14	0.3291(4)	0.36	0.36	0.19
Fit: [8, 15]									
m_V	0.8621(176)	0.8746(58)	0.33	0.21	0.11	0.8738(34)	0.20	0.11	0.06

is between 0.6 and 3 using the fitting range as shown in Tables IV and V. The quantity r_{Cost} defined in Eqs. (27) and (26) indicates the computational cost compared to the standard method, with and without deflation, respectively. Comparing costs for masses of the nucleon, N^*, and vector mesons with LMA and AMA, one sees that error reduction in AMA is much larger than from LMA at both $m = 0.005$ and $m = 0.01$. AMA has a cost reduction for those observables of about 5 to 20 times larger compared to the standard method and LMA. It can easily be understood by looking at r_{Error} of those hadron masses in AMA which is close to the ideal ratio ($1/\sqrt{N_G} \simeq 0.18$), and the construction cost of $\mathcal{O}^{(\text{appx})}$ is much cheaper than the original one. In particular, for the N^*, the gain from AMA compared to LMA is even more dramatic. Actually, in

LMA, the Δr term dominates the total error in Eq. (14), and it turns out that error reduction by LMA is limited to $\sqrt{2\Delta r}$ even if N_G is increased to $N_G = V$, as is usually done. Improvement for heavy mesons and baryons would also be interesting work.

Considering the multiple-source method with deflation, statistics are increased by averaging over hadron propagators with N_{src} different source locations. In such a case, the original cost is given by the CG cost times N_{src} plus the cost of generating eigenvectors,

$$\text{Cost}_{\text{w/defl}}(\text{multisource})$$
$$= (\#\text{Mult}_{\text{deflCG(org)}} \times 12 \times N_{\text{src}} + \#\text{Mult}_{\text{Lanczos}}) \times N_{\text{conf}}. \tag{29}$$

TABLE V. Same as shown in Table IV at $m = 0.01$.

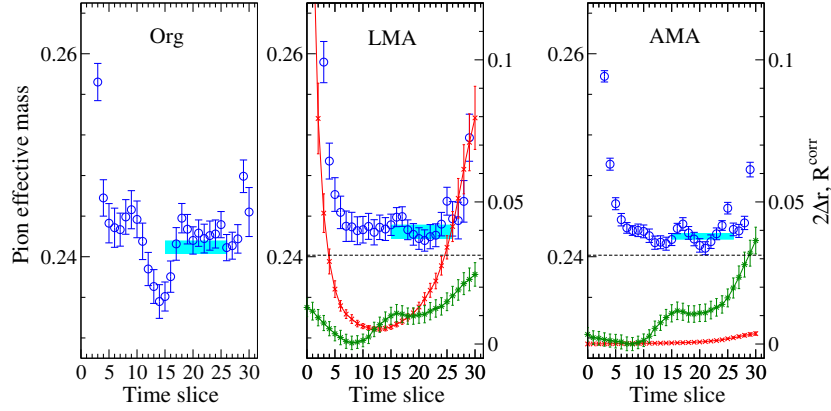| $m = 0.01$									
	Org	LMA	r_{Error}	$r_{\text{Cost}}^{\text{w/o defl}}$	$r_{\text{Cost}}^{\text{w/defl}}$	AMA	r_{Error}	$r_{\text{Cost}}^{\text{w/o defl}}$	$r_{\text{Cost}}^{\text{w/defl}}$
Fit: [7, 12]									
m_N	1.2279(127)	1.2234(63)	0.50	0.51	0.25	1.2422(24)	0.19	0.14	0.07
$E_N(n_p^2 = 1)$	1.2877(156)	1.2992(76)	0.49	0.49	0.24	1.3222(27)	0.17	0.12	0.06
$E_N(n_p^2 = 2)$	1.3438(207)	1.3682(97)	0.47	0.46	0.22	1.3981(32)	0.16	0.09	0.05
$E_N(n_p^2 = 3)$	1.3695(289)	1.4256(145)	0.50	0.52	0.25	1.4677(45)	0.16	0.09	0.05
$E_N(n_p^2 = 4)$	1.4661(437)	1.4944(206)	0.47	0.46	0.22	1.5379(63)	0.15	0.08	0.04
Fit: [5, 8]									
m_{N^*}	1.800(49)	1.659(69)	1.40	4.02	1.95	1.787(11)	0.23	0.20	0.10
Fit: [15, 26]									
m_π	0.4169(10)	0.4195(11)	1.08	2.41	1.17	0.4187(4)	0.47	0.83	0.40
Fit: [8, 15]									
m_V	0.9185(124)	0.9228(67)	0.55	0.62	0.30	0.9198(29)	0.24	0.22	0.11

FIG. 6 (color online).   The effective mass plot of the pion at $m = 0.01$, and $2\Delta r$ and R^{corr} of the pion propagator as in Fig. 5.

Assuming that there is no correlation between different source locations, we can set $N_{\mathrm{src}} = N_G$, so the reduction of computational cost is

$$r_{\mathrm{Cost}}(\text{multisource}) = \frac{\mathrm{Cost(AMA)}}{\mathrm{Cost_{w/defl}}(\text{multisource})}$$

$$\simeq 0.49(m = 0.005),$$

$$\simeq 0.33(m = 0.01). \qquad (30)$$

The computational cost advantage of AMA is cut in half compared to the case with no deflation. However, this relative cost will decrease again if additional propagators are computed, for instance, for three-point functions (see next section), or if the lattice size is increased and more source translations are used.

In the case of the pion, comparing r_{Error} in LMA between $m = 0.005$ and $m = 0.01$, we find Δr at $m = 0.01$ is much larger than at $m = 0.005$. This is due to less dominance of the low modes and the use of fewer low modes in our setup at $m = 0.01$: the approximation is worse as seen in Figs. 5 and 6. Using AMA, thanks to the relaxed CG, the approximation is improved. We also notice that r_{Error} for the pion mass is about 1.5 times larger than for the pion propagator (see Fig. 2 and Table IV). This is due to the relatively large value of R^{corr} for a pion propagator above $t = 16$. This observation is confirmed if we extend the distance between $\mathcal{O}^{(\mathrm{appx})g}$ and $\mathcal{O}^{(\mathrm{appx})g'}$. For example, using source shifts only in the temporal direction (source

separation in the temporal direction is longer than in the spatial direction), $N_G = 4$, r_{Error} of the pion mass is similar to the ideal, $1/\sqrt{N_G} = 0.5$, as shown in Table VI. It turns out that for the pion the correlation R^{corr} is relatively significant in the error reduction rate.

D. Nucleon form factors

In this section we apply AMA to nucleon three-point functions which have a more complicated structure in terms of quark propagators. We carry out the measurement of three-point functions [(nucleon)-(operator)-(nucleon)] where the operators are vector (V_μ) or axial-vector (A_μ) currents, and we evaluate the axial-charge and isovector form factors defined from the matrix elements,

$$\langle N_1(p_1, s)|V_\mu^a|N_0(p_0, s)\rangle$$
$$= \bar{u}_{N_1}(p_1, s)\left[\gamma_\mu F_1^a(q^2) + \frac{\sigma_{\mu\nu}q_\nu}{2m_N}F_2^a(q^2)\right]u_{N_0}(p_0, s),$$
$$(31)$$

$$\langle N_1(p_1, s)|A_\mu^a|N_0(p_0, s)\rangle$$
$$= \bar{u}_{N_1}(p_1, s)[\gamma_\mu\gamma_5 F_A^a(q^2) + iq_\mu\gamma_5 F_2^a(q^2)]u_{N_0}(p_0, s),$$
$$(32)$$

with momenta $\vec{p}_0$ and $\vec{p}_1$ of on-shell nucleon states N_0 and N_1, respectively, with spin s. The superscript "a" is an SU (2) flavor index referring to either isovector or isoscalar

TABLE VI.   Pion and vector meson mass as shown in Table IV at $m = 0.005$ and $N_G = 4$.

$m = 0.005$								
	LMA	r_{Error}	$r_{\mathrm{Cost}}^{\mathrm{w/o\,defl}}$	$r_{\mathrm{Cost}}^{\mathrm{w/defl}}$	AMA	r_{Error}	$r_{\mathrm{Cost}}^{\mathrm{w/o\,defl}}$	$r_{\mathrm{Cost}}^{\mathrm{w/defl}}$
Fit: [15, 26]								
m_π	0.3286(6)	0.52	0.52	0.27	0.3287(6)	0.51	0.53	0.28
Fit: [8, 15]								
m_V	0.8840(94)	0.54	0.55	0.29	0.8801(83)	0.47	0.45	0.23

components. Below we study matrix elements of the isovector currents ($a = +$). F_1^a and F_2^a are obtained from the Sachs form factors,

$$G_E(q^2) = F_1^a(q^2) - \frac{q^2}{4m_N^2} F_2^a(q^2),$$

$$G_M(q^2) = F_1^a(q^2) + F_2^a(q^2). \tag{33}$$

The isovector form factor $F_A^+(q^2)$ at zero momentum transfer is known as the axial charge of the nucleon, $g_A = F_A^+(0)$, which is an important quantity governing neutron β decay.

To obtain the form factors, we construct ratios of three-point correlation functions, $C_{J_\mu}^N$, and nucleon two-point functions, $C_{G,L}^N$, as

$$R_{J_\mu}(t_1, t, t_0 | p_1, p_0)$$
$$= K \frac{C_{J_\mu}^N(\vec{q}, t)}{C_G^N(t_1 - t_0, 0)}$$
$$\times \left[\frac{C_L^N(t_1 - t, \vec{q}) C_G^N(t - t_0, 0) C_L^N(t_1 - t_0, 0)}{C_L^N(t_1 - t, 0) C_G^N(t - t_0, \vec{q}) C_L^N(t_1 - t_0, \vec{q})} \right]^{1/2} \tag{34}$$

with $K = \sqrt{2(E_N + m_N)/E_N}$, where C_L^N is with the point-sink and gauge-invariant Gaussian smeared source, and C_G^N is with the gauge-invariant Gaussian smeared source and sink. t_0, t_1 denote the temporal location of the initial and final states of nucleon which are fixed, and t is the temporal location of the operator which moves between t_0 and t_1. The momentum transfer is defined as $q = p_0 - p_1$, and in our setup we use $p_0 = (E_N, \vec{p})$ and $p_1 = (m_N, 0)$ with $\vec{p} = (p_x, p_y, p_z) = 2\pi\vec{n}_p/L$, $\vec{n}_p^2 = 0, \ldots, 4$. In order to extract the form factors of the ground state nucleon from R_{J_μ} we use the spin-projection matrix $P_4 = (1 + \gamma_4)/2$, and $P_{5z} = P_4\gamma_5\gamma_3$, as in [31]. For the vector case,

$$\lim_{t_1 - t, t - t_0 \gg 1} \mathrm{tr}[P_{5z}(R_{V_1} + R_{V_2})](t_1, t, t_0 | p_1, p_0)$$
$$= \frac{-ip_x + ip_y}{m_N} G_M(q^2), \tag{35}$$

$$\lim_{t_1 - t, t - t_0 \gg 1} \mathrm{tr}[P_4 R_{V_4}](t_1, t, t_0 | p_1, p_0) = \frac{E_N + m_N}{m_N} G_E(q^2), \tag{36}$$

and for the axial vector,

$$\lim_{t_1 - t, t - t_0 \gg 1} \mathrm{tr}[P_{5z}(R_{A_1} + R_{A_2})](t_1, t, t_0 | p_1, p_0)$$
$$= -\frac{(p_x + p_y)p_z}{m_N} F_P(q^2), \tag{37}$$

$$\lim_{t_1 - t, t - t_0 \gg 1} \mathrm{tr}[P_{5z}R_{A_3}](t_1, t, t_0 | p_1, p_0)$$
$$= \frac{1}{m_N}[m_N F_A(q^2) - p_z^2 F_P(q^2)], \tag{38}$$

after taking $t_1 \gg t \gg t_0$ to project on the nucleon ground state. In the above derivation we use the normalization for Dirac spinors, $\sum_s \bar{u}_N(p, s) u_N(p, s) = 2m_N$. The parameters of the gauge-invariant Gaussian smeared source sink are the same as in [31], and $t_0 = 0$, $t_1 = 12$. In this calculation we employ the local currents $V_\mu^a = \bar{q}\gamma_\mu\tau^a q$ and $A_\mu^a = \bar{q}\gamma_\mu\gamma_5\tau^a q$ where τ^a is flavor SU(2) generator normalized as $\mathrm{tr}\tau^a\tau^b = \delta^{ab}$, and hence we multiply matrix elements of the currents by the renormalization constant $Z_V = 0.7178$, determined nonperturbatively [26].

We compare the axial charge and isovector form factor at each momentum between the standard method and LMA or AMA. Figure 7 shows g_A for two different masses. A ground state plateau is clearly observed for $4 \le t \le 8$ for both masses. Comparing the contribution of Δr and R^{corr} in LMA and AMA, one sees that Δr in AMA is much smaller, and the quality of the approximation is significantly enhanced. In cost estimates of the three-point functions, we compute "polarized" and "unpolarized" matrix elements for both up-type and down-type contractions which is an additional cost factor of four quark propagators. As shown in Tables VII and VIII, AMA achieves error reductions in G_A, F_1^+, and F_2^+ close to $1/\sqrt{N_G}$ with 5–20 times smaller computational cost than the standard method or LMA. Comparing the results for AMA at the two masses $m = 0.005$ and $m = 0.01$, the error reduction compared to the standard method is significant for both, despite having fewer eigenvectors for the latter. The cost ratios, comparing to the multisource method with $N_{\mathrm{src}} = N_G$, are

$$r_{\mathrm{Cost}}(\text{multisource}) \simeq 0.32 (m = 0.005),$$
$$\simeq 0.24 (m = 0.01), \tag{39}$$

in which we have gains greater than factors of 3 and 4 for AMA. We also note that not only have the statistical errors decreased dramatically, but the plateaus are much more readily observed for AMA.

IV. FUTURE EXTENSION

This paper has shown numerical tests of AMA using the relaxed CG as the approximation, but there are many other examples of $\mathcal{O}^{(\mathrm{appx})}$. One idea is to employ improved DWF actions, e.g., Möbius-type [34] or Borici-type [35,36], which are extensions of DWF allowing smaller L_s without enhancing chiral symmetry breaking, in addition to the relaxed CG solver. Such improvements have other benefits like the reduction of memory or disk-storage size of eigenvector data stored on the disk.
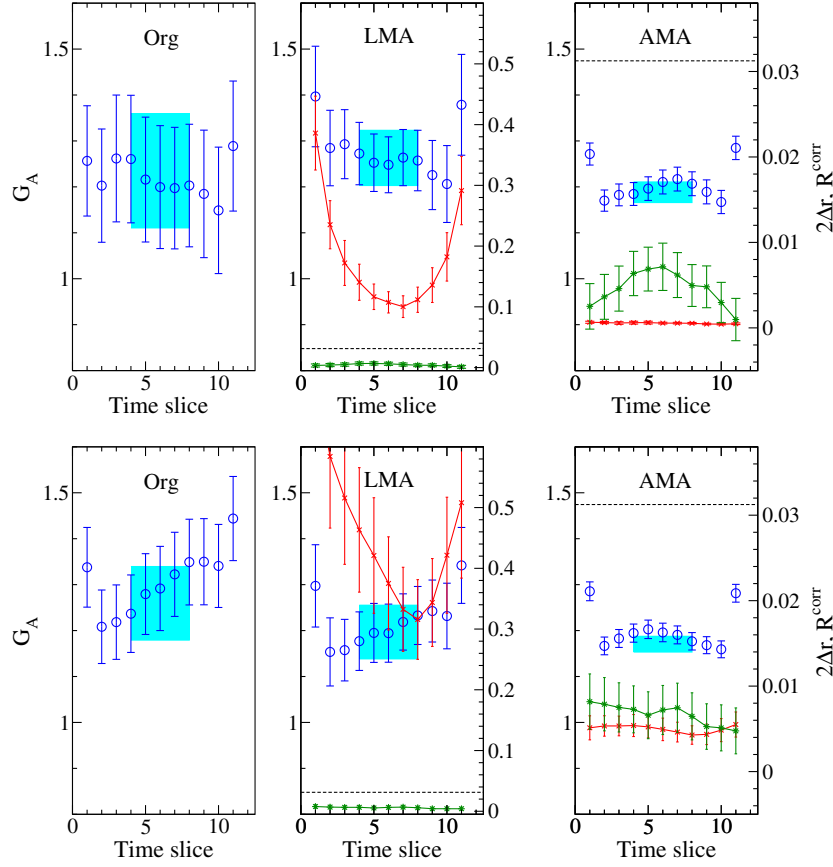
FIG. 7 (color online).   Time-slice dependence of axial-charge G_A in $m = 0.005$ (top) and $m = 0.01$ (bottom) with standard method (left), LMA (middle) and AMA (right). The cross symbols and star symbols denote $2\Delta r$ and R^{corr} for the three-point function in Eq. (34). The colored band is the constant fitting result in this range.

We test the above strategy on another DWF ensemble generated by the RBC/UKQCD Collaboration [37], with larger lattice size ($32^3 \times 64$) and $L_s = 32$, and smaller pion mass, $m_\pi \approx 170$ MeV. For the approximation we take a Möbius-type DWF Dirac operator with $L_s = 16$. We use 1000 low modes, computed with a 200 degree Chebychev polynomial, and then only 2 restarts of the Lanczos procedure are needed. In this case, the computational cost ratio reads

$$
\begin{aligned}
\mathrm{Cost(AMA)} = &(\#\mathrm{Mult}_{\mathrm{Lanczos}} \times 0.6 + (\#\mathrm{Mult}_{\mathrm{deflCG(org)}} \\
&+ \#\mathrm{Mult}_{\mathrm{deflCG(AMA)}} \times 0.6 N_G) \times 12) \times N_{\mathrm{conf}},
\end{aligned}
\tag{40}
$$

where the factor 0.6 arises from the fact that there is an additional 20% cost for the multiplication with the Möbius-type Dirac operator compared to a DWF operator with the same L_s length together with the having of the cost due to using $L_s/2$ for the Möbius-type Dirac operator, i.e., $1.2/2 = 0.6$. The axial charge is shown in Fig. 8. One sees that there is a clear plateau between 3 and 6, where we set the source and sink operator at time slice 0 and 9, respectively, and around the plateau the correlation Δr has a similar order as for the $m = 0.01$, $24^3 \times 64$ case discussed in the last section. In Tables IX and X we summarize hadron masses and the axial charge for both the standard method and AMA. From those tables, the ratio of errors is close to the ideal one, $1/\sqrt{112} \simeq 0.094$, and thus $\mathcal{O}^{(\mathrm{appx})}$ is

TABLE VII.   Axial charge G_A with standard method, LMA, and AMA in $m = 0.005$ and $m = 0.01$.

G_A	Org	LMA	r_{Error}	$r_{\mathrm{Cost}}^{\mathrm{w/o\,defl}}$	$r_{\mathrm{Cost}}^{\mathrm{w/defl}}$	AMA	r_{Error}	$r_{\mathrm{Cost}}^{\mathrm{w/o\,defl}}$	$r_{\mathrm{Cost}}^{\mathrm{w/defl}}$
Fit: [4, 8]									
$m = 0.005$	1.235(124)	1.263(60)	0.48	0.11	0.23	1.188(22)	0.18	0.04	0.09
$m = 0.01$	1.259(80)	1.197(58)	0.73	0.35	0.53	1.170(17)	0.21	0.11	0.17

TABLE VIII. F_1^+ and F_2^+ with standard method, LMA, and AMA in $m = 0.01$.

	Org	LMA	r_{Error}	$r_{\text{Cost}}^{\text{w/o defl}}$	$r_{\text{Cost}}^{\text{w/defl}}$	AMA	r_{Error}	$r_{\text{Cost}}^{\text{w/o defl}}$	$r_{\text{Cost}}^{\text{w/defl}}$
Fit: [4, 8]									
$F_1^+(n_p^2 = 1)$	0.849(53)	0.860(52)	0.99	0.64	0.97	0.799(10)	0.20	0.10	0.15
$F_1^+(n_p^2 = 2)$	0.695(50)	0.730(47)	0.95	0.60	0.91	0.678(10)	0.20	0.10	0.15
$F_1^+(n_p^2 = 3)$	0.493(57)	0.618(47)	0.82	0.45	0.68	0.583(11)	0.21	0.10	0.16
$F_1^+(n_p^2 = 4)$	0.406(50)	0.524(49)	0.97	0.62	0.94	0.555(17)	0.35	0.30	0.45
$F_2^+(n_p^2 = 1)$	2.61(26)	2.35(17)	0.66	0.28	0.43	2.37(5)	0.19	0.09	0.13
$F_2^+(n_p^2 = 2)$	1.88(22)	1.91(14)	0.66	0.29	0.44	1.85(4)	0.19	0.09	0.13
$F_2^+(n_p^2 = 3)$	1.52(16)	1.62(13)	0.82	0.44	0.67	1.52(4)	0.25	0.15	0.23
$F_2^+(n_p^2 = 4)$	1.12(15)	1.17(13)	0.86	0.49	0.74	1.32(5)	0.35	0.29	0.44

still a good approximation to the original even though we use Möbius-type DWF. AMA reduces the computational cost by 10 to 30 times in this case.

Still other approximations are possible. For instance, the inexactly deflated CG, using the EigCG algorithm [38] with low precision, is adopted as $\mathcal{O}^{(\text{appx})}$. This uses low-precision eigenmodes as well as deflation, and will be beneficial for long-distance observables corresponding to pion and kaon physics. Especially for large lattice sizes, since there are many available source locations, it is possible to reduce the size of gauge ensembles while still maintaining statistical precision. Furthermore we also note that in [19] the hopping parameter expansion for the inverse of the Dirac matrix is used as the approximation $\mathcal{O}^{(\text{appx})}$. These are a few of the new directions to pursue high precision calculations without additional computational cost in a Monte Carlo simulation (however, a careful analysis of autocorrelation times is necessary).

V. DISCUSSION AND SUMMARY

As shown in the previous section, AMA is a powerful tool for the precise measurement of observables obtained from correlation functions in Monte Carlo simulations.

Defining the improved estimator $\mathcal{O}^{(\text{imp})}$ using the approximation $\mathcal{O}^{(\text{appx})}$, which has the same covariance properties as the original $\mathcal{O}$ but a much smaller construction cost, $\mathcal{O}^{(\text{imp})}$ has smaller statistical errors without additional computational cost. In this paper we employ the relaxed CG with deflation to produce the approximation. Since the computational cost of the approximation using the relaxed CG is much less than the original one, the observables needing many quark propagators with the CG solve of the Dirac matrix benefit accordingly from the AMA method. Figures 9, 10, and 11 show the ratio of computational costs for AMA. One sees that, compared to the propagator, the cost of the CG solves for the nucleon form factor dominates the total cost. This is because 4 extra CG solves are necessary to construct the three-point functions. Figure 12 shows the summary of the reduction rate of computational cost for LMA and AMA as in Tables IV, V, VII, IX, and X. The computational cost of G_A in AMA is reduced more than the two-point function, and also AMA has an advantage of more than 7 times speed-up for computation of two- and three-point functions compared to the traditional method. We also notice that, for $32^3 \times 64$ lattice size and DSDR gauge action ("32cID"), there is
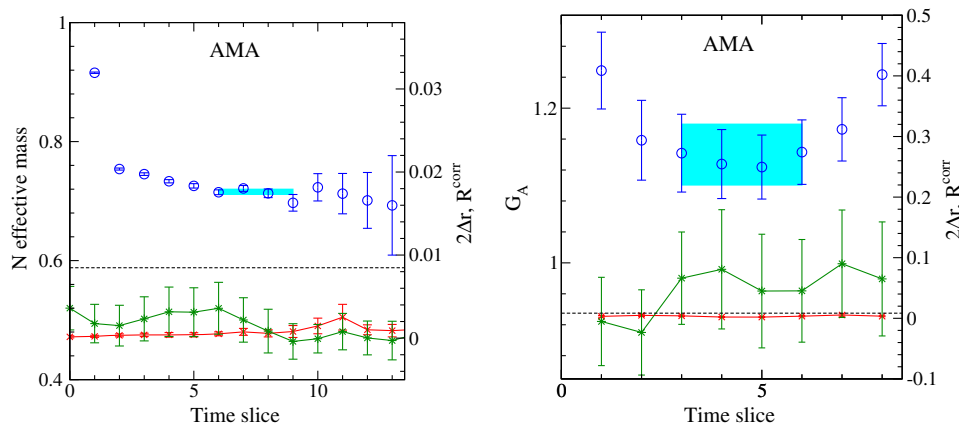

FIG. 8 (color online).   The nucleon effective mass plot and the axial charge, as shown in Fig. 7, at $m = 0.001$ with Iwasaki + DSDR action in $32^3 \times 64$ lattice.

TABLE IX.   Result of hadron mass in the DSDR lattice in $m = 0.001$.

	Org	AMA	r_{Error}	$r_{\text{Cost}}^{\text{w/o defl}}$
Fit: [6, 9]				
m_N	0.9625(538)	0.9822(57)	0.11	0.04
$E_N(n_p^2 = 1)$	0.9759(524)	1.0201(59)	0.11	0.04
$E_N(n_p^2 = 2)$	1.0090(515)	1.0568(65)	0.13	0.06
$E_N(n_p^2 = 3)$	1.0466(509)	1.0900(74)	0.15	0.08
$E_N(n_p^2 = 4)$	1.0035(544)	1.1268(84)	0.16	0.08
Fit: [4, 7]				
m_N	1.445(258)	1.430(24)	0.09	0.03
Fit: [8, 21]				
m_π	0.1694(21)	0.1712(3)	0.18	0.11
Fit: [6, 10]				
m_V	0.8502(821)	0.7414(77)	0.09	0.03

TABLE X.   Result of G_A in DSDR lattice in $m = 0.001$.

	Org	AMA	r_{Error}	$r_{\text{Cost}}^{\text{w/o defl}}$
Fit: [3, 6]				
G_A	1.401(275)	1.135(42)	0.15	0.05

more than 10 times reduction of $r_{\text{Cost}}^{\text{w/o defl}}$. by employing the Möbius operator in the approximation. There are also realistic DWF simulations at the physical quark mass point with 5.5 fm volume with two lattice spacings, which employed AMA [39]. It turns out that AMA also works well for an approximation which is made from a different action than the original one. As shown in Fig. 11, the computational cost of a precise CG solve with DWF is still large, in fact, 29% for the propagator and 46% for the form factor, since we did not use the deflation method in the original one. Further cost reduction by applying the modified deflation method in CG with Möbius DWF eigenmodes is currently under way [40].
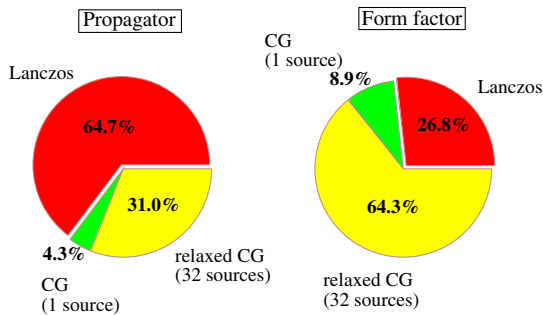
FIG. 9 (color online).   The rate of computational cost of AMA for the hadron propagator (left) and three-point function of the form factor (right) at $m = 0.005$. This is in the case of 400 eigenmodes computation and the use of 32 source locations for relaxed CG ($\varepsilon = 0.003$) in AMA.
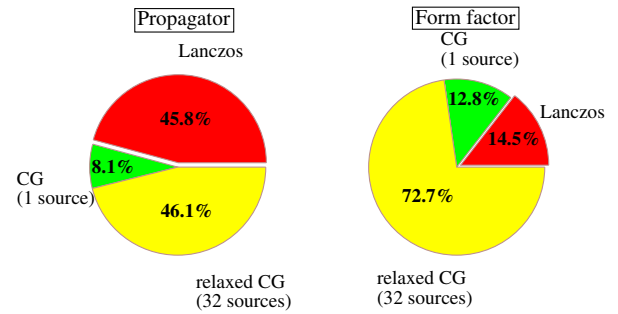
FIG. 10 (color online).   Same as Fig. 9 at $m = 0.01$. This is in the case of 180 eigenmodes computation and the use of 32 source locations for relaxed CG ($\varepsilon = 0.003$) in AMA.
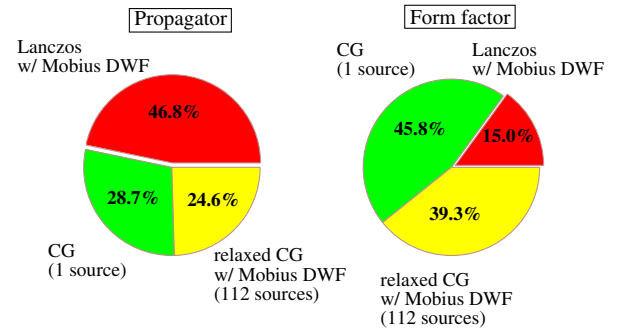
FIG. 11 (color online).   Same as Fig. 9 for the $32^3 \times 64 \times 32$ DSDR lattice. This is in the case of 1000 eigenmodes computation and the use of 112 source locations for relaxed CG with Möbius DWF kernel in AMA.

We comment on the relation of the approximation with the low-mode distribution of the Dirac operator. As in Eqs. (18) and (19), the deflation with low modes increases the quality of the approximation since these are treated exactly in the inverse of the Dirac operator. However, in this case there appears the additional computational cost of the eigenvectors. So in AMA we need to find the appropriate
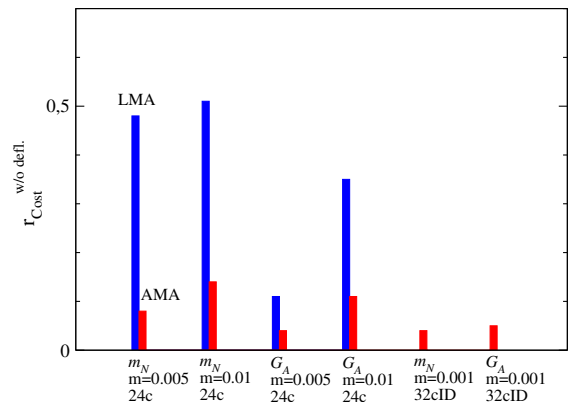
FIG. 12 (color online). $r_{\text{Cost}}^{\text{w/o defl}}$ of nucleon mass m_N and axial charge G_A for LMA (blue bar) and AMA (red bar).
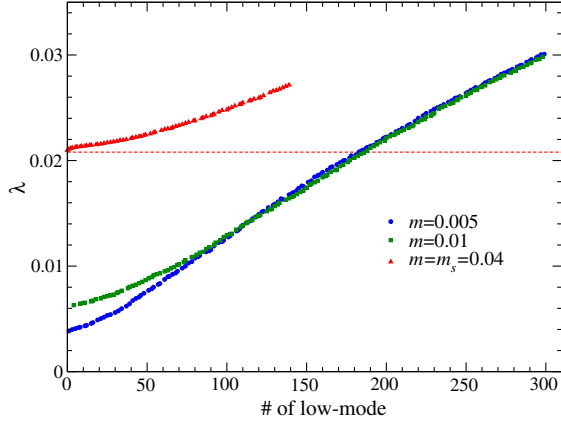
FIG. 13 (color online). Distribution of positive low-lying eigenvalue at the light quark mass $m = 0.005, 0.01$ and the strange quark mass $m = m_s = 0.04$. The dashed line shows the lowest eigenvalue for the strange quark mass.

value of N_λ by considering a balance between additional eigenmode cost and benefit for deflation. In the DWF case, the benefit of deflation in a strange quark mass regime is much less than in a light quark mass regime. As shown in Fig. 13, one sees that the lowest eigenvalue of the strange quark Dirac operator has a similar magnitude as in the $N_\lambda = 180$ point in both $m = 0.005$ and $m = 0.01$. It turns out that the approximation for the strange quark without deflation has a similar gain as in the light quark mass with $N_\lambda = 180$. We know that AMA with $N_\lambda = 180$ in $m = 0.01$ has a certain cost reduction for two- and three-point functions, and thus, at the strange quark mass, AMA without low-mode deflation also has an advantage.

AMA is an example of a new class of CAA which reduces the statistical error on correlation functions in Monte Carlo simulations in an efficient way. Although AMA is similar to LMA, we have shown that it works not only for low-mode dominated observables (associated with the pion) but also for a broad range of observables involving baryons and other mesons by taking account of contributions from all modes of the Dirac operator. In AMA we have used the CG inverter with a relaxed stopping criterion as the approximation, and we numerically tested this method in lattice QCD with $N_f = 2 + 1$ dynamical DWF on lattice sizes of $24^3 \times 64$ and $L_s = 16$ and inverse spacing $a^{-1} = 1.73$ GeV. Our tests correspond to pions with masses in the range 300 to 500 MeV. Using AMA, we have shown reductions of computational cost of more than 5 times compared to the standard method for nucleon and vector meson masses, the axial charge and isovector form factors of the nucleon. These results suggest interesting applications to observables having long-standing hurdles of large statistical noise to precise measurements, e.g., the neutron electric dipole moment, muon anomalous magnetic moment, and proton decay matrix elements [41]. The application of AMA to all of these is now under way.

APPENDIX A: STANDARD DEVIATION OF THE IMPROVED ESTIMATOR

The standard deviation of the improved estimator in (8) is given as

$$\sigma^{(\text{imp})} = \sqrt{\langle (\Delta \mathcal{O}^{(\text{imp})})^2 \rangle}. \tag{A1}$$

Here we express the correlation between $\mathcal{O}$, $\mathcal{O}^{(\text{appx})}$, and $\mathcal{O}^{(\text{appx})g}$ as

$$r_g = \frac{\langle \Delta \mathcal{O} \Delta \mathcal{O}^{(\text{appx})g} \rangle}{\sigma \sigma_g^{(\text{appx})}}, \tag{A2}$$

$$r_g^{\text{corr}} = \frac{\langle \Delta \mathcal{O}^{(\text{appx})} \Delta \mathcal{O}^{(\text{appx})g} \rangle}{\sigma^{(\text{appx})} \sigma_g^{(\text{appx})}}, \tag{A3}$$

$$r_{gg'}^{\text{corr}} = \frac{\langle \Delta \mathcal{O}^{(\text{appx})g} \Delta \mathcal{O}^{(\text{appx})g'} \rangle}{\sigma_g^{(\text{appx})} \sigma_{g'}^{(\text{appx})}}, \tag{A4}$$

where, if g is the unit transformation I, we have $r_I = r$ and $r_{Ig'}^{\text{corr}} = r_{g'}^{\text{corr}}$. Substituting (A2), (A3), and (A4) into (7) and (8), we have

$$\sigma^{(\text{imp})} = \left[\sigma^2 - 2r\sigma^{(\text{appx})}\sigma + \sigma^{(\text{appx})2} \right.$$
$$+ \frac{2}{N_G} \sum_g \sigma_g^{(\text{appx})} (r_g \sigma - r_g^{\text{corr}} \sigma^{(\text{appx})})$$
$$\left. + \frac{1}{N_G^2} \left(\sum_g \sigma_g^{\text{appx}2} + \sum_{g \neq g'} \sigma_g^{(\text{appx})} \sigma_{g'}^{(\text{appx})} r_{gg'}^{\text{corr}} \right) \right]^{1/2}. \tag{A5}$$

Assuming that the standard deviation of $\mathcal{O}$ is equivalent with $\mathcal{O}^{(\text{appx})}$,

$$\sigma^{(\text{appx})} \simeq \sigma_g^{(\text{appx})} \simeq \sigma, \qquad (A6)$$

we have

$$\sigma^{(\text{imp})} \simeq \sigma \left[2(1-r) + \frac{2}{N_G} \sum_g (r_g - r_g^{\text{corr}}) \right.$$
$$\left. + \frac{1}{N_G^2} \left(N_G + \sum_{g \neq g'} r_{gg'}^{\text{corr}} \right) \right]^{1/2}. \qquad (A7)$$

Furthermore if the correlation between $\mathcal{O}^{(\text{appx})}$ and $\mathcal{O}^{(\text{appx})g}$ is negligibly small,

$$r_g^{\text{corr}} \simeq 0, \qquad r_{gg'}^{\text{corr}} \simeq 0, \qquad r_g \simeq 0 \qquad (A8)$$

(the last one assumes the correlation between $\mathcal{O}^{(\text{appx})g}$ and $\mathcal{O}$ is small), we have

$$\sigma^{(\text{imp})} \simeq \sigma \sqrt{2(1-r) + \frac{2r-1}{N_G}}. \qquad (A9)$$

APPENDIX B: NOTE ON POSSIBLE BIAS DUE TO ROUND-OFF ERROR

In this section, we address the possible appearance of bias due to the round-off error for finite machine precision. Although the AMA estimator does not have any bias if the exact arithmetic is carried out, it is important to notice whether a significant breaking of covariant symmetry by round-off error appears. We strongly advise that, in practice, one should explicitly check that the size of the bias is negligible on a few configurations as is done below (Fig. 16), or follow the method in Appendix C to remove the bias completely.

There are two possible sources. One is, only when a fixed norm of the residual vector in the CG is used as the stopping condition in the approximation part of the improved estimator, the difference of CG iteration rarely occurring in a verge of stopping condition because of inexact arithmetic of residual vector-norm computation. Second is round-off error accumulating in the iterative solver algorithm at the arithmetic step of multiplication of vector-vector and vector-matrix. In our numerical study, however, we show it does not appear even in sub-% precision.

Here the bias is defined as the violation of the equivalence Eq. (3),

$$\langle \mathcal{O}^g[U] \rangle = \langle \mathcal{O}[U^g] \rangle + \delta_{\mathcal{O}}, \qquad (B1)$$

where $\delta_{\mathcal{O}} \neq 0$ indicates the amount of systematic error. This is a consequence of the breaking of covariance in Eq. (4),

$$\mathcal{O}^g[U] \neq \mathcal{O}[U^g]. \qquad (B2)$$

This breaking may not be negligible when a very crude approximation is employed, or accumulation of machine epsilon is somehow enhanced under weak circumstances for the round-off effect.

1. Threshold error in fixed stopping condition for residual vector

In the following, we show the first example of bias effect and numerical check. This is only the most obvious place where small differences due to the finite precision matters. When we use the CG for the construction of f_ε in the second term of Eq. (18), the accuracy of f is measured by using the residual vector r defined as the difference between the source vector and matrix H times the approximation vector f, $r = b - Hf$. Its norm corresponds to the accuracy of f, f_ε, and it is given as the sum over lattice sites,

$$\|r\|^2 = \sum_x r^\dagger(x) r(x) = r^\dagger(x_1) r(x_1) + r^\dagger(x_2) r(x_2) + \cdots$$
$$+ r^\dagger(x_V) r(x_V). \qquad (B3)$$

We notice that the above norm is slightly different from the one resulting if the right-hand side of Eq. (B2) is computed instead, due to the order of arithmetic,

$$\|r^g\|^2 = \sum_{x=x^g} r^\dagger(x) r(x)$$
$$= r^\dagger(x_1 + \delta) r(x_1 + \delta) + r^\dagger(x_2 + \delta) r(x_2 + \delta) + \cdots$$
$$+ r^\dagger(x_V + \delta) r(x_V + \delta)$$
$$\neq \|r\|^2, \qquad (B4)$$

where g denotes the transformation for $x^g = x + \delta$ with constant shift vector δ. When the stopping condition ε used in AMA falls between $\|r^g\|$ and $\|r\|$, the number of CG iterations is different,

$$N_{\text{CG}}(\|r\|) \neq N_{\text{CG}}(\|r^g\|), \qquad (B5)$$

which leads to the breaking of Eq. (B2). This discrepancy affects Eq. (19),

$$f_\varepsilon(H[U](x,y)) = \sum_{k=1}^{N_{\text{CG}}(\|r\|)} c_k[U](H[U])^k(x,y), \qquad (B6)$$

where $N_{\text{CG}}(\|r\|)$, the number of CG iterations when the fixed stopping condition of the norm of the residual vector is used. $c_k[U]$ is a coefficient implicitly determined by the CG procedure. Because of Eq. (B5), the discrepancy of the CG part under the transformation g arises as
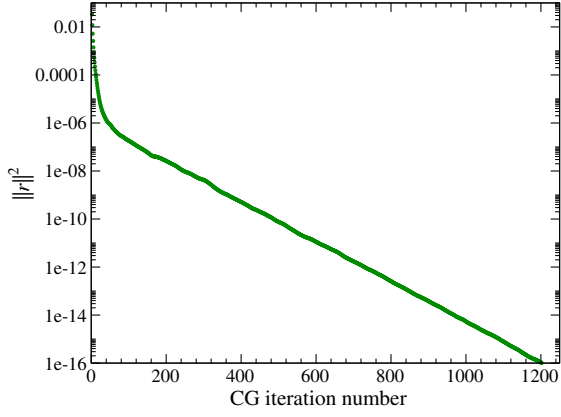
FIG. 14 (color online). The relation between the squared norm of the residual vector and CG iteration number.

$$f_\varepsilon(H[U](x^g, y^g)) = \sum_{k=1}^{N_{CG}(\|r\|)} c_k[U]H^k[U](x^g, y^g)$$

$$= \sum_{k=1}^{N_{CG}(\|r^g\|)} c_k[U^g]H^k[U^g](x, y)$$

$$+ \Delta_f = f_\varepsilon(H[U^g](x, y)) + \Delta_f \quad \text{(B7)}$$

[here we assume that $c_k[U]H^k[U](x^g, y^g) = c_k[U^g] \times H^k[U^g](x, y)$ within machine precision]. Δ_f does not vanish when a different number of iteration by round-off error accidentally appears as in Eq. (B5). Therefore there is no guarantee of cancellation between $\langle \mathcal{O}^{AMA} \rangle$ and $\langle \mathcal{O}_G^{AMA} \rangle$. This breaking may be significant if a very low precision for the stopping condition is chosen, where f_ε rapidly changes for the initial CG iterations. For example, as seen from Fig. 14, when the CG iteration number is changed from $N_{CG} = 20$ to 21, the accuracy of solution vector changes by the order $\|r(x)\| \simeq 10^{-3}$. On the other hand, in the region of $N_{CG} = 1200$, even if N_{CG} is changed from 1200 to 1201, the accuracy of the solution vector is still less than $\|r(x)\| \simeq 10^{-9}$, and it turns out that the effect of different N_{CG} of relaxed CG in $\mathcal{O}^{(appx)}$ is more significant than N_{CG}

Algorithm 1. CG algorithm for solving $Ax_{CG} = b$ with positive Hermite matrix A.

1: *if* $k := 0$ *then*
2: $x_0 := 0$
3: $r_0 := b - Ax_0$, $p_0 := r_0$
4: *end if*
5: *while* $\|r_k\| > \epsilon$ *do*
6: $\alpha_k := \frac{(r_k, r_k)}{(p_k, Ap_k)}$
7: $x_{k+1} := x_k + \alpha_k p_k$, $r_{k+1} := r_k - \alpha_k Ap_k$
8: $\beta_k := \frac{(r_{k+1}, r_{k+1})}{(r_k, r_k)}$
9: $p_{k+1} := r_{k+1} + \beta_k p_k$
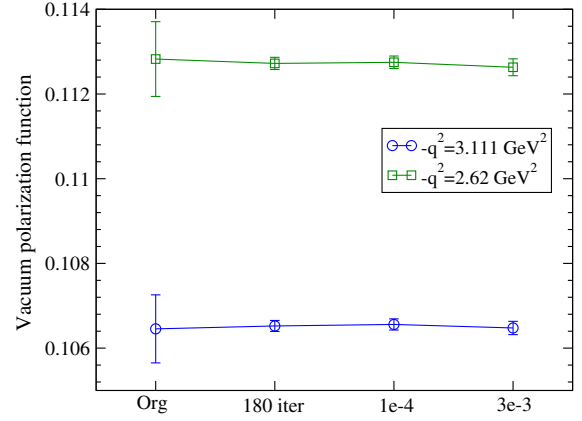10: $k := k + 1$
11: *end while*

FIG. 15 (color online). The vacuum polarization function of vector type using the same gauge configurations at $m = 0.005$. The number of configurations are 51. "Org" denotes the results without AMA; "1e-4," "3e-3," and "180 iter" denote the AMA results using 10^{-4}, 3×10^{-3} stopping criteria of the norm of residual vector and 180 CG iteration, respectively. The different symbols are results in different $-q^2$ points of vacuum polarization function.

of exact CG in $\mathcal{O}$ (and also such bias is totally suppressed within machine precision for $\mathcal{O}$). Obviously this kind of bias does not appear when f_ε is constructed by a fixed CG iteration number instead of a fixed norm of residual vector as the stopping condition.

In Fig. 15 we numerically compare the result of the vacuum polarization function (VPF) with two procedures of AMA used in the 3×10^{-3} and 10^{-4} stopping condition for the norm of residual vector and 180 CG iterations. The VPF is extracted from the conserved vector and local vector current correlator following [42–44]. One sees that the resulting values of the VPF from two different stopping conditions are consistent within statistical error whose accuracy is at the sub-percent level. This result supports that the systematic error of arithmetic bias addressed in this section is not visible in the practical calculations. Note that the mechanism that enhances the size of the bias due to the threshold effect of the residual vector norm mentioned above is avoided when using the fixed CG iteration number.

2. Accumulated round-off error

The round-off error due to inexact arithmetic in an iterative solver could potentially destroy the covariance that is crucial for AMA and introduce bias. Below we show in a realistic case that the round-off error is innocuous. CAA conceptually relies on preserving the covariant symmetry in each iteration, e.g., from step 6 to step 9 in Algorithm 1. After many vector-vector and matrix-vector multiplies to determine the residual and search vectors, the accumulation of round-off error due to the different orders of arithmetic may spoil the exact covariant symmetry.
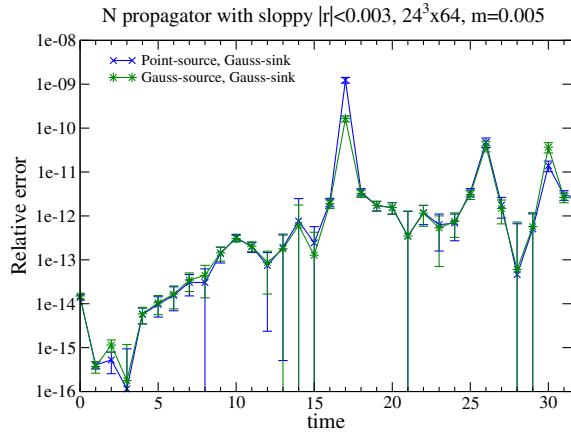
FIG. 16 (color online). Relative error of δ_c for nucleon propagator with Gaussian-source and point sink (cross) and Gaussian-source and Gaussian-sink (star) as a function of time slice. This is averaged once using 16 source locations on one reference configuration with low-mode deflation using 400 low modes at $m = 0.005$ in the $24^3 \times 64$ lattice.

The extent to which the symmetry is violated, of course, depends too on the details of the algorithm.[4]

To check the preservation of covariance in the AMA approximation, we compare nucleon two-point correlation functions with those computed after translating the position of both the nucleon source and the gauge links. If the floating point arithmetic were exact, the nucleon correlation functions would have to be identical, which means the bias in AMA is zero. The bias caused by the finite precision arithmetic is quantified as

$$\delta_c = \mathcal{O}^{(\text{appx})g}[U^{\bar{g}}] - \mathcal{O}^{(\text{appx})}[U], \qquad (B8)$$

where g denotes the transformation, and $\bar{g}$ denotes the inverse transformation of g. In our test the source position and link variables are shifted using 16 different translations, $(12, 0, 0, 0)$, $(0, 12, 0, 0)$, …, $(12, 12, 12, 32)$ on one configuration. The only difference with the original unshifted calculation is the order of arithmetic in the Lanczos and CG algorithm according to the shift of the gauge configuration and fermion source point. In Fig. 16, one sees that the effect of round-off error on the covariant symmetry, when using the low-mode deflation with 400 low-lying eigenmodes as used in the present work, is $O(10^{-9})$ (and much smaller in the part of the correlation function that is statistically well resolved) and does not depend on smeared or local source type. Thus the approximation with sloppy CG using the 0.003 residual stopping condition is not significantly affected by accumulative

---

[4]For example, the BiCG-type algorithm which is much less stable than CG may be more susceptible to accumulated effects of round-off. We thank T. Doi for pointing this out to us after making a test with Wilson-clover fermions.

round-off errors, and hence systematic bias. In fact, even if such a round-off error did introduce a bias due to the relative order of arithmetic, it can be removed by the technique explained in the next section which does not rely on covariance.

APPENDIX C: ERROR REDUCTION TECHNIQUE WITHOUT COVARIANT SYMMETRY

In this section we introduce another estimator in which the random transformation $g_r \in G_r$ is adopted for $\mathcal{O}^{(\text{appx})}$ instead of covariance. Employing g_r, which is assumed as the element of group G_r, into Eq. (8), the improved estimator is defined as

$$\mathcal{O}^{(\text{imp})g_r} = \mathcal{O}^{g_r} - \mathcal{O}^{(\text{appx})g_r} + \mathcal{O}_G^{(\text{appx})g_r}, \qquad (C1)$$

$$\mathcal{O}_G^{(\text{appx})g_r} = \frac{1}{N_G} \sum_{g \in G} \mathcal{O}^{(\text{appx})g \circ g_r}. \qquad (C2)$$

The second equation has the multitransformation $g \circ g_r$ with g and g_r for $\mathcal{O}^{(\text{appx})}$. Here we also assume G as the subset of G_r.

We prove that this estimator does not have any bias provided the numerical procedure of $\mathcal{O}^{(\text{appx})}$ is deterministic and reproducible; these calculations are bit-by-bit the same for the same input parameters (gauge configuration, source location, stopping criteria, etc.). We note that our program is always checked to reproduce bit-by-bit the same results for the same input. Since the biasless estimator should satisfy the equivalence of the expectation value as

$$\langle \mathcal{O} \rangle = \langle \mathcal{O}^{g_r} \rangle = \langle \mathcal{O}^{(\text{imp})g_r} \rangle \qquad (C3)$$

(here we consider $\mathcal{O}$ is covariant under g_r), thus, from Eq. (C1) and by using the transformation of the link variable with g_r, we show

$$\langle \mathcal{O}^{(\text{appx})g_r} \rangle = \langle \mathcal{O}_G^{(\text{appx})g_r} \rangle, \qquad (C4)$$

even if $\mathcal{O}^{(\text{appx})}$ does not follow from a covariant symmetry. In the above, the expectation value is defined as the group integral of link variables and the summation over $g_r \in G_r$.

The left-hand side of Eq. (C4) is described as

$$\langle \mathcal{O}^{(\text{appx})g_r} \rangle = \frac{1}{Z} \sum_{g_r \in G_r} P(g_r) \int dU e^{-S[U]} \mathcal{O}^{(\text{appx})g_r}[U], \qquad (C5)$$

where $S[U]$ denotes the QCD action, and $P(g_r)$ denotes the distribution function of $g_r \in G_r$ normalized to unity. Z is the partition function. On the other hand, the right-hand side of Eq. (C4) can be written as

$$\langle \mathcal{O}_G^{(\mathrm{appx})g_r} \rangle = \frac{1}{Z} \sum_{g_r \in G_r} P(g_r) \frac{1}{N_G} \sum_{g \in G} \int dU e^{-S[U]} \mathcal{O}^{(\mathrm{appx})g \circ g_r}[U].$$

(C6)

Here we consider that the multiplication of $g_r \in G_r$ with $g \in G$ is also an element of G_r, i.e., $g \circ g_r \in G_r$, and the distribution function of $g \circ g_r$ is the same function of $g_r \in G_r$, i.e., $P(g \circ g_r) = P(g_r)$, when $G \subseteq G_r$. In this case, Eq. (C6) can be expressed as a single sum over $g_r \in G_r$, and so its equation is equivalent to Eq. (C5). We notice that in this derivation it is unnecessary to use the covariance of $\mathcal{O}^{(\mathrm{appx})}$. Practically g_r is chosen randomly in each configuration, for instance, a random shift of source location for $\mathcal{O}$ and $\mathcal{O}^{(\mathrm{appx})}$. Hence, to avoid any bias due to the arithmetic error explained in Appendix B, $\mathcal{O}^{(\mathrm{imp})g_r}$ instead of $\mathcal{O}^{(\mathrm{imp})}$ is appropriate when the CG stopping condition is chosen as the fixed norm of the residual vector. Note that, in Eq. (C1), g_r is only performed for each functional; the link variables are not transformed. When the link variable is transformed instead of $\mathcal{O}^{(\mathrm{appx})}$, the biasless of $\mathcal{O}^{(\mathrm{imp})g_r}$ is only guaranteed for $\mathcal{O}^{(\mathrm{appx})}$ by the covariance under G and G_r.

APPENDIX D: IMPLICITLY RESTARTED LANCZOS ALGORITHM WITH POLYNOMIAL ACCELERATION

Suppose that $A \in \mathbb{C}^{N \times N}$ is the Hermitian, positive definite, matrix. Introducing the tridiagonal matrix $T \in \mathbb{C}^{m \times m}$ whose diagonal and off-diagonal components are $\alpha_{i=1,\dots,m}$ and $\beta_{i=1,\dots,m-1}$, respectively, the relation

$$AV = VT + r_m e_m^\dagger \qquad (D1)$$

provides T and the orthogonal matrix $V \in \mathbb{C}^{N \times m}$ recursively as shown in Algorithm 3. In the above equation e_m denotes the unit vector with a nonzero value in the mth component. If $V^\dagger r_m \simeq 0$, the $k(\leq m)$th eigenvector ψ_k and eigenvalue (λ_k) of matrix A are given by the multiplication of the unitary matrix obtained by the diagonalization for tridiagonal matrix, $T = U^\dagger \Lambda U$, as $UV = \{\psi_k\}, \Lambda = \mathrm{diag}(\lambda_k)$.

The restarted Lanczos algorithm is based on the concept to recycle the final vector v_m in the Lanczos iteration as the new initial vector v^{new} in order to avoid the storage constraints. Suppose that m is divided into k *wanted* eigenvectors $\{v_1, \dots, v_k\}$ which is the desired region of the eigenvalue distribution, and p *unwanted* vectors $\{v_{k+1}, \dots, v_{k+p}\}$ which are recomputed in every step of the Lanczos iteration after restarting. After running $m \equiv k + p$ Lanczos steps, we restart the Lanczos process with initial vector and β value,

$$v_{k+1}^{\mathrm{new}} = v_m, \qquad \beta_k' = \beta_m, \qquad (D2)$$

Algorithm 2. QR factorization process.

1: Let set $T_1 = T$ and $i = 1$
2: *while* $i = p$ *do*
3: $\quad T_i - \mu_i = Q_i R_i$
4: $\quad R_i Q_i + \mu_i = T_{i+1}$
5: $\quad i = i + 1$
6: *end while*

---

and thus the orthogonal matrix V is constructed by

$$V = \{v_1, \dots, v_k\} \cup \{v_{k+1}^{\mathrm{new}}, \dots, v_m^{\mathrm{new}}\} \subset \{v_1, \dots, v_{m+p}\}.$$

(D3)

Effectively after the restarted Lanczos step we obtain vectors v_i spanning the Krylov space $K_{m+p}(A, v_1)$. The last equation in (D3) may be broken due to round-off errors, leading to loss of orthogonality in the restarted process, since it does not take account of reorthogonalization with previous *unwanted* vectors $\{v_{k+1}, \dots, v_m\}$. Such an effect, however, depends on the choice of p, and in the actual lattice QCD simulation, the less than 5 times restarted Lanczos process has no matter of orthogonality loss.

Usually we implement the filtering technique using QR factorization and shifting the resulting tridiagonal matrix. In this algorithm we employ the approximate *unwanted* eigenvalues as shift parameters $\mu_i = \tilde{\lambda}_{i=k+1,\dots,m}$ and obtain the orthogonal matrix $Q = \prod_{i=1}^p Q_i$ from the QR factorization process (see Algorithm 2).

$V_+ = VQ$ and T_p are also satisfied with the Lanczos recursion relation

$$(AV_+)_{ij} = (V_+ T_p)_{ij} + (r_m)_i Q_{mj}, \qquad V_+ = VQ, \quad (D4)$$

and thus the new initial vector v_{k+1}^{new} alternative to Eq. (D2) consists of

$$r_{k+1}^{\mathrm{new}} = v_{k+1}^+ + Q_{mk} r_m, \qquad \beta_{k+1}^{\mathrm{new}} = \|r_{k+1}^{\mathrm{new}}\|,$$
$$v_{k+1}^{\mathrm{new}} = r_{k+1}^{\mathrm{new}} / \beta_{k+1}^{\mathrm{new}}, \qquad (D5)$$

with rotated vector $v_i^+ = \sum_{l=1}^m Q_{li} v_l$ $[i = 1, \dots, k+1]$. In the above we use the relation of $T_{i+1} = Q_i^\dagger T_i Q_i$ and $Q_{m,i} = 0$ $[i = 1, \dots, k-1]$. Therefore we can restart the Lanczos step from $k+1$ to $k+p$ following Algorithm 3, and we generate the new orthogonal matrix

$$V^{\mathrm{new}} = \{v_1^+, \dots, v_k^+\} \cup \{v_{k+1}^{\mathrm{new}}, \dots, v_m^{\mathrm{new}}\}, \quad (D6)$$

which also spans the Krylov space $\mathcal{K}_{m+p}(A, v_1)$. Note that via QR factorization the new *wanted* vector $v_{1,\dots,k}^+$ is automatically multiplied by the filtering polynomial function

$$f_p(A) = \prod_{i=k+1}^{m} (A - \tilde{\lambda}_i), \qquad \text{(D7)}$$

and thus

$$v_i^+ \propto f_p(A) v_i, \qquad \text{(D8)}$$

which is known from the relation of $V_+ e_1 = VQe_1 \propto f_p(A) v_1$. The filtering polynomial function may suppress the *unwanted* vectors. Fulfilling the *unwanted* eigenvalue constraints on $|f(\lambda_{i=k+1,...,m})| < |f(\lambda_k)|$, the polynomial function of Eq. (D8) works as a filter of *unwanted* eigenmodes from the spectrum of A [28,29].

The restarted Lanczos algorithm combined with polynomial acceleration [27] emphasizes the low-lying *wanted* eigenvectors in the Krylov space and suppresses the *unwanted* vector via the filtering function. Let us consider the computation of the low modes of Hermitian matrix H whose maximum absolute eigenvalue is already known as $\lambda_{\max}$. The Chebychev polynomial function T_{chev} can be used to easily control the eigenvalue distribution of H by enhancing the wanted small eigenvalue region ($\lambda < \alpha$) and suppressing the unwanted region. By applying T_{chev} with the following argument function:

$$q(H) = \frac{2H^2 - \alpha^2 - \beta^2}{\beta^2 - \alpha^2}, \qquad \text{(D9)}$$

we have that

$$|T_{\text{chev}}^n(q(\lambda))| \gg 1, \qquad \lambda^2 \notin [\alpha^2, \beta^2],$$
$$T_{\text{chev}}^n(q(\lambda)) \in [-1, 1], \qquad \lambda^2 \in [\alpha^2, \beta^2], \qquad \text{(D10)}$$

where we set α slightly larger than the maximum *wanted* eigenvalue, and $\beta^2 \geq \lambda_{\max}^2$ (see Fig. 17). $T_{\text{chev}}^n(q(H))$, constructed by a recursion relation,
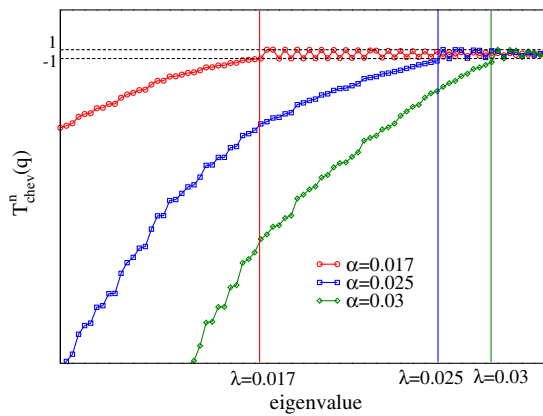

FIG. 17 (color online). The sketch of Chebychev polynomial $T_{\text{chev}}^n(q)$ as a function of eigenvalue. Different symbols illustrate the $T_{\text{chev}}^n(q)$ with several α.

$T_{\text{chev}}^n(x) = 2x T_{\text{chev}}^{n-1}(x) - T_{\text{chev}}^{n-2}(x)$, has the same eigenvectors as H, and the highest eigenvalue of $T_{\text{chev}}^n(q(H))$ corresponds to the lowest eigenvalue of H. The degree n of T_{chev}^n, which is also the number of its zeros in $[-1, 1]$, depends on the magnitude of the highest eigenvalue and the hierarchy of magnitudes for the *wanted* eigenvalues. Recalling the restarted Lanczos process, if we set α close to the lowest point in the eigenvalue region $\lambda_{i=k+1,...,m}$, the filtering function in Eq. (D8) strongly suppresses the *unwanted* eigenvalue region.

We easily extend the polynomial acceleration techniques to focus on an arbitrary range of *wanted* eigenvalues by introducing the shift parameter μ into Eq. (D9),

$$q(H, \mu) = \frac{2(H - \mu)^2 - \alpha^2 - (\beta + |\mu|)^2}{(\beta + |\mu|)^2 - \alpha^2}, \qquad \text{(D11)}$$

in which this argument function enhances the spectrum in the range $\lambda = (\mu - \alpha, \mu + \alpha)$.

APPENDIX E: 4D EVEN-ODD PRECONDITIONING IN DOMAIN-WALL FERMIONS

In this appendix we explicitly present the definition of DWF 4D even-odd preconditioning (see [45] and [34] and references therein) which is used not only in the preconditioning of the CG solver but also in the computation of eigenvectors and eigenvalues in the Lanczos algorithm. Instead of DWF five-dimensional (5D) even-odd preconditioning as has been used in [26], the DWF operator can be expressed as the even-odd hopping matrix in 4D space-time in which the Wilson-fermion kernel of DWF is in the off-diagonal blocks and the 5D hopping term is in diagonal blocks of the following matrix:

$$
\begin{aligned}
D_{\text{DW}}((x, s), (y, t)) &= (5 - M_5)[\delta_{x,y} W_5(s, t) \\
&\quad - K W_4(x, y) \delta_{s,t}] \\
&= (2K)^{-1} \begin{pmatrix} I_{ee} W_5 & -K W_{4eo} \\ -K W_{4oe} & I_{oo} W_5 \end{pmatrix}
\end{aligned}
$$
$$\text{(E1)}$$

Algorithm 3. Lanczos algorithm.

1: Set v_1 to the unit vector, $\beta_0 = 0$ and $k = 0$;
2: *while* $k = m$ *do*
3: $\quad \alpha_k = (v_k, A v_k)$;
4: $\quad r_k = (A - \alpha_k) v_k - \beta_{k-1} v_{k-1}$;
5: $\quad \beta_k = \|r_k\|$;
6: $\quad v_{k+1} = r_k / \beta_k$;
7: $\quad$ Gram-Schmidt reorthogonalization for $v_1, ..., v_{k+1}$ if we
$\quad\quad$ needed;
8: $\quad k = k + 1$;
9: *end while*

---

in which we use

$$K = \frac{1}{2(5 - M_5)}, \tag{E2}$$

$$W_4(x, y) = \sum_\mu [(1 + \gamma_\mu) U^\dagger(x - \hat\mu)\delta_{x-\hat\mu,y} + (1 - \gamma_\mu) U(x)\delta_{x+\hat\mu,y}], \tag{E3}$$

$$W_5(s, t) = 1 - 2K(P_R \delta_{s,t+1} + P_L \delta_{s+1,t} - m P_R \delta_{s,1}\delta_{t,L_s} - m P_L \delta_{s,L_s}\delta_{t,1}), \tag{E4}$$

with SU(3) link variable $U_\mu(x)$ and Dirac γ-matrix. Here we suppress color and spin indices in the DWF operator. Evenness or oddness of a site of Euclidean space-time is given as $\mathrm{mod}(\sum_{\mu=1}^4 x_\mu, 2) = 0$ or 1. M_5 is the so-called domain wall height.

The inverse of the DWF operator in even-odd representation is expressed through the Schur decomposition as

$$D_{\mathrm{DW}}^{-1} = (2K)^{-1} \begin{pmatrix} I_{ee} & 0 \\ K W_5^{-1} W_{4eo} & I_{oo} W_5^{-1} \end{pmatrix} \begin{pmatrix} D_{ee}^{-1} & 0 \\ 0 & I_{oo} \end{pmatrix} \begin{pmatrix} I_{ee} & K W_{4eo} W_5^{-1} \\ 0 & I_{oo} \end{pmatrix}, \tag{E5}$$

$$D_{ee} = I_{ee} W_5 - K^2 W_{4eo} W_5^{-1} W_{4oe}, \tag{E6}$$

in which the inverse of W_5 can be represented explicitly,

$$W_5^{-1}(s, t) = A(s, t) P_R + B(s, t) P_L, \tag{E7}$$

$$A(s, t) = \delta_{st} - \frac{1}{1 + m\kappa^{L_s}} \begin{pmatrix} m\kappa^{L_s} & m\kappa^{L_s - 1} & m\kappa^{L_s - 2} & \cdots & m\kappa \\ -\kappa & m\kappa^{L_s} & m\kappa^{L_s - 1} & \cdots & m\kappa^2 \\ -\kappa^2 & -\kappa & m\kappa^{L_s} & \cdots & m\kappa^3 \\ \vdots & \vdots & \vdots & & \vdots \\ -\kappa^{L_s - 1} & -\kappa^{L_s - 2} & -\kappa^{L_s - 3} & \cdots & m\kappa^{L_s} \end{pmatrix}_{st}, \tag{E8}$$

$$B(s, t) = A(t, s), \tag{E9}$$

with $\kappa = (5 - M_5)^{-1}$.

In a practical implementation of W_5^{-1}, it is convenient to use the LU decomposition. Using the left and right representations of W_5,

$$W_5(s, t) = P_R [I - \kappa(\Delta + \Delta_m)]_{st} + P_L [I - \kappa(\Delta^T + \Delta_m^T)]_{st} \tag{E10}$$

with

$$\Delta = \begin{pmatrix} 0 & & & & & & 0 \\ 1 & 0 & & & & & \\ & 1 & 0 & & & & \\ & & \ddots & \ddots & & & \\ & & & & 1 & 0 & \\ 0 & & & & & 1 & 0 \end{pmatrix}, \qquad \Delta_m = \begin{pmatrix} 0 & & -m \\ & \ddots & \\ 0 & & 0 \end{pmatrix}. \tag{E11}$$

We also know that the matrix without P_R is represented as

$$[I - \kappa(\Delta + \Delta_m)] = (1 - \kappa\Delta_m(I - \kappa\Delta)^{-1})(I - \kappa\Delta), \tag{E12}$$

and

$$(I - \kappa\Delta)^{-1} = \begin{pmatrix} 1 & & & & \\ k & 1 & & & \\ \vdots & & \ddots & & \\ \kappa^{L_s-2} & \kappa^{L_s-3} & \cdots & \kappa & 1 \\ \kappa^{L_s-1} & \kappa^{L_s-2} & \cdots & \kappa^2 & \kappa & 1 \end{pmatrix}. \tag{E13}$$

Thus we have

$$[I - \kappa(\Delta + \Delta_m)] = () \begin{pmatrix} 1 + m\kappa^{L_s} & m\kappa^{L_s-1} & m\kappa^{L_s-2} & \cdots & m\kappa \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & \ddots & \\ & & & 1 & 0 \\ 0 & & \cdots & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & & 0 & & \\ -\kappa & 1 & 0 & & \\ & \ddots & \ddots & \ddots & \\ & & -\kappa & 1 & 0 \\ & & & -\kappa & 1 \end{pmatrix}. \tag{E14}$$

Finally we obtain

$$A(s,t) = \begin{pmatrix} 1 & & & & \\ \kappa & 1 & & & \\ \vdots & & \ddots & & \\ \kappa^{L_s-2} & \kappa^{L_s-3} & \cdots & \kappa & 1 \\ \kappa^{L_s-1} & \kappa^{L_s-2} & \cdots & \kappa^2 & \kappa & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{1+m\kappa^{L_s}} & \frac{-m\kappa^{L_s-1}}{1+m\kappa^{L_s}} & \frac{-m\kappa^{L_s-2}}{1+m\kappa^{L_s}} & \cdots & \frac{-m\kappa}{1+m\kappa^{L_s}} \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & \ddots & \\ & & & 1 & 0 \\ 0 & & \cdots & 0 & 1 \end{pmatrix}. \tag{E15}$$

Now the number of floating-point operations in the multiplication of $A(s,t)$ with a vector is reduced to $(L_s^2 - 1)/2$ from L_s^2, i.e., a gain of $(L_s^2 + 1)/2$.

γ_5-Hermiticity of the DWF operator is given by

$$D_{\mathrm{DW}}^\dagger(s,t) = \sum_{s_1,t_1} \Gamma_5(s,s_1) D_{DW}(s_1,t_1) \Gamma_5(t_1,t), \tag{E16}$$

with $\Gamma_5(s,t) = \gamma_5 \delta_{s,L_s-t+1}$; hence the Hermiticity of the even-odd preconditioned Domain-wall operator

$$H_{ee} = \Gamma_5 D_{ee} \tag{E17}$$

follows from D_{ee}, $D_{ee}^\dagger = \Gamma_5 D_{ee} \Gamma_5$, since Γ_5 is a diagonal matrix at each 4D even-odd site. The difference from DWF 5D even-odd preconditioning is that H_{ee} can be represented as a single multiplication of Γ_5 without a flip of the even-odd site. Equation (E17) can be used in the Lanczos algorithm with $H = H_{ee}$ in Eqs. (D9) and (D11).

[1] E. Shintani, S. Aoki, N. Ishizuka, K. Kanaya, Y. Kikukawa, Y. Kuramashi, M. Okawa, Y. Taniguchi, A. Ukawa, and T. Yoshié, Phys. Rev. D **72**, 014504 (2005).

[2] F. Berruto, T. Blum, K. Orginos, and A. Soni, Phys. Rev. D **73**, 054509 (2006).

[3] E. Shintani, S. Aoki, N. Ishizuka, K. Kanaya, Y. Kikukawa, Y. Kuramashi, M. Okawa, A. Ukawa, and T. Yoshié, Phys. Rev. D **75**, 034507 (2007).

[4] E. Shintani, S. Aoki, and Y. Kuramashi, Phys. Rev. D **78**, 014503 (2008).

[5] T. Blum, M. Hayakawa, and T. Izubuchi, *Proc. Sci.*, LATTICE2012 (2012) 022 [arXiv:1301.2607].

[6] N. Christ, C. Dawson, T. Izubuchi, C. Jung, Q. Liu, R. D. Mawhinney, C. T. Sachrajda, A. Soni, and R. Zhou, Phys. Rev. Lett. **105**, 241601 (2010).

[7] T. Blum, T. Izubuchi, and E. Shintani, Phys. Rev. D **88**, 094503 (2013).

[8] L. Giusti, C. Hoelbling, M. Luscher, and H. Wittig, Comput. Phys. Commun. **153**, 31 (2003).

[9] L. Giusti, P. Hernandez, M. Laine, P. Weisz, and H. Wittig, J. High Energy Phys. 04 (2004) 013.

[10] T. A. DeGrand and S. Schaefer, Comput. Phys. Commun. **159**, 185 (2004).

[11] T. A. DeGrand and S. Schaefer, Phys. Rev. D **72**, 054503 (2005).

[12] M. Luscher, J. High Energy Phys. 12 (2007) 011.

[13] H. Fukaya et al. (JLQCD Collaboration), Phys. Rev. Lett. **98**, 172001 (2007).

[14] J. Noaki et al. (JLQCD and TWQCD Collaborations), Phys. Rev. Lett. **101**, 202004 (2008).

[15] L. Giusti and S. Necco, Proc. Sci., LAT2005 (2006) 132 [arXiv:hep-lat/0510011].

[16] A. Li et al. (xQCD Collaboration), Phys. Rev. D **82**, 114501 (2010).

[17] G. Bali, L. Castagnini, and S. Collins, Proc. Sci., LATTICE2010 (2010) 096 [arXiv:1011.1353].

[18] M. Gong et al., Phys. Rev. D **88**, 014503 (2013).

[19] G. S. Bali, S. Collins, and A. Schafer, Comput. Phys. Commun. **181**, 1570 (2010).

[20] J. Kolorenc and L. Mitas, Rep. Prog. Phys. **74**, 026502 (2011).

[21] L. Pollet, Rep. Prog. Phys. **75**, 094501 (2012).

[22] W. Foulkes, L. Mitas, R. Needs, and G. Rajagopal, Rev. Mod. Phys. **73**, 33 (2001).

[23] J. Carlson, S. Gandolfi, and A. Gezerlis, Prog. Theor. Exp. Phys. **2012**, 01A209 (2012).

[24] W. Leidemann and G. Orlandini, Prog. Part. Nucl. Phys. **68**, 158 (2013).

[25] T. Blum, T. Izubuchi, and E. Shintani, Proc. Sci., LATTICE2012 (2012) 262 [arXiv:1212.5542].

[26] Y. Aoki et al. (RBC Collaboration, UKQCD Collaboration), Phys. Rev. D **83**, 074508 (2011).

[27] Y. Saad, Math. Comput. Modell. **42**, 567 (1984).

[28] D. C. Sorensen, SIAM J. Matrix Anal. Appl. **13**, 357 (1992).

[29] D. Calvetti, L. Reichel, and D. C. Sorensen, Electron. Trans. Numer. Anal. **2**, 21 (1994).

[30] H. Neff, N. Eicker, T. Lippert, J. W. Negele, and K. Schilling, Phys. Rev. D **64**, 114509 (2001).

[31] T. Yamazaki, Y. Aoki, T. Blum, H.-W. Lin, S. Ohta, S. Sasaki, R. Tweedie, and J. Zanotti, Phys. Rev. D **79**, 114505 (2009).

[32] S. Sasaki, T. Blum, and S. Ohta, Phys. Rev. D **65**, 074503 (2002).

[33] H.-W. Lin, Chin. J. Phys. (Taipei) **49**, 827 (2011).

[34] R. C. Brower, H. Neff, and K. Orginos, arXiv:1206.5214.

[35] A. Borici, Nucl. Phys. B, Proc. Suppl. **83**, 771 (2000).

[36] A. Borici, arXiv:hep-lat/0402035.

[37] R. Arthur et al. (RBC Collaboration, UKQCD Collaboration), Phys. Rev. D **87**, 094514 (2013).

[38] A. Abdel-Rehim, A. Stathopoulos, and K. Orginos, Linear Algebra Appl. **21**, 473 (2014).

[39] T. Blum et al., Proc. Sci., LATTICE2013 (2014) 404.

[40] H. Yin and R. D. Mawhinney, Proc. Sci., LATTICE2011 (2011) 051 [arXiv:1111.5059].

[41] Y. Aoki, E. Shintani, and A. Soni, Phys. Rev. D **89**, 014505 (2014).

[42] E. Shintani, S. Aoki, T. W. Chiu, S. Hashimoto, T. H. Hsieh, T. Kaneko, H. Matsufuru, J. Noaki, T. Onogi, and N. Yamada (JLQCD Collaboration, TWQCD Collaboration), Phys. Rev. D **79**, 074510 (2009).

[43] P. Boyle, L. Del Debbio, E. Kerrane, and J. Zanotti, Phys. Rev. D **85**, 074504 (2012).

[44] C. Aubin and T. Blum, Phys. Rev. D **75**, 114502 (2007).

[45] A. Pochinsky, http://www.mit.edu/avp/mdwf/.