

Computational Physics

Accelerating multigrid with streaming chiral SVD for Wilson fermions in lattice QCD

Travis Whyte^{a,*}, Andreas Stathopoulos^b, Eloy Romero^c^a Jülich Supercomputing Center, Forschungszentrum Jülich, Jülich, 52428, Germany^b Department of Computer Science, William & Mary, Williamsburg, Virginia, USA^c Thomas Jefferson National Accelerator Facility, Newport News, Virginia, USA

ARTICLE INFO

Editor: Prof. Z. Was

Keywords:

Lattice quantum chromodynamics

Multigrid

Iterative methods

ABSTRACT

A modification to the setup algorithm for the multigrid preconditioner of Wilson fermions in lattice QCD is presented. A larger basis of test vectors than that used in regular multigrid is calculated by the smoother and truncated by singular value decomposition on the chiral components of the test vectors. The truncated basis is used to form the prolongation and restriction matrices of the multigrid hierarchy. This modification of the setup method is demonstrated to increase the convergence of linear solvers on an anisotropic lattice with $m_\pi \approx 239$ MeV from the Hadron Spectrum Collaboration and an isotropic lattice with $m_\pi \approx 220$ MeV from the MILC Collaboration. The lattice volume dependence of the method is also examined. Increasing the number of test vectors improves speedup up to a point, but storing these vectors becomes impossible in limited memory resources such as GPUs. To address storage cost, we implement a *streaming* singular value decomposition of the basis of test vectors on the chiral components and demonstrate a decrease in the number of fine level iterations by a factor of 1.7 for $m_q \approx m_{crit}$.

1. Introduction

Quantum Chromodynamics (QCD) is the theory of the strong interaction, which governs the dynamics of quarks and gluons. A non-perturbative approach to QCD is *lattice* QCD, wherein the equations governing the strong interaction are discretized to a hypercubic spacetime lattice where they can be solved numerically. One of the most computationally expensive parts of any lattice QCD calculation is the generation of *propagators*, which are created by solving systems of linear equations

$$Dx = b \quad (1)$$

where $D \in \mathbb{C}^{n \times n}$ is the lattice Dirac operator, a large sparse matrix arising from the discretization of the quark and gluon fields. This matrix has a dependence on the underlying background gauge field, quark mass and volume of the spacetime lattice. As the mass of the light quarks approaches their physical value, the eigenspectrum of the lattice Dirac operator approaches the origin of the complex plane. In this regime, iterative solvers experience *critical slowing-down*, wherein the number of iterations required for the system of linear equations to reach convergence begins to drastically increase as the lowest eigenvalues of the matrix approach zero. Critical slowing-down is also experienced as the volume of the lattice grows larger, which results in increased density of

the matrix spectrum near zero. This increased density near the origin of the complex plane further increases the difficulty of solving the system of linear equations, and preconditioning methods of the iterative solvers are required.

Many preconditioning methods have been developed and used within lattice QCD to combat the increasing computational cost of generating propagators. Such methods include even-odd preconditioning, ILU, additive and multiplicative Schwarz, deflation, polynomial preconditioning and most recently, multigrid; see a good introduction to these techniques at [1]. In the first multigrid attempts for lattice QCD, a renormalization group approach was utilized to construct the multigrid hierarchy (see a review of early work at [2]). While the multigrid preconditioner created in this approach exhibited speedup compared to regular methods, critical slowing down was not fully eliminated on the finest grid. It was not until the advent of adaptive multigrid [3] that a successful multigrid preconditioner for Wilson fermions was created. In adaptive multigrid, the components of the error that the smoother cannot sufficiently reduce, i.e. the low frequency components, are exposed by using the smoother to create a set of algebraically smooth near null or test vectors. These test vectors are rich in components in the direction of the low-lying eigenspace and are used to create a coarse grid matrix through a projection of the original fine grid matrix. This process

* Corresponding author.

E-mail address: t.whyte@fz-juelich.de (T. Whyte).<https://doi.org/10.1016/j.cpc.2026.110089>

Received 21 May 2025; Received in revised form 13 February 2026; Accepted 19 February 2026

Available online 12 March 2026

0010-4655/© 2026 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

can be performed recursively to create matrices associated with coarser grids.

The number of test vectors used to create the coarse grid matrices must be chosen judiciously. A large number of test vectors results in coarse grid matrices that are efficient at reducing the error associated with the low lying eigenspace but are expensive to apply. This is because the degrees of freedom at each grid point of the coarser grid, and by extension the number of non-zeros of the matrix, scale quadratically with the number of test vectors. Consequently, there is an overall increase in the time to solution due to the cost of the stencil application and the time spent communicating the enlarged halos [4]. Conversely, having too few test vectors leads to small coarse grid matrices that are inexpensive to apply. However, the resulting preconditioner may not adequately reduce the error associated with the low-lying eigenspace, which results in slower convergence. We are motivated in looking for an improvement in the multigrid setup process that can achieve the convergence properties of a multigrid preconditioner created with a large number of test vectors while maintaining the cost of one constructed with a smaller number of test vectors.

Since Ref [3], multigrid preconditioners for other fermion discretizations have been created, eg. Brower et al. [5,6]. However, little work has been done to improve the multigrid setup process for Wilson fermions beyond its initial implementation. Most studies have focused on specific parts of the setup process, such as smoothers [7,8], reducing the condition number of the coarse grid matrix [9,10], or replacing the multigrid preconditioner with a deep neural network [11]. In this article, a modification of the setup method for the multigrid preconditioner of Wilson type fermions is presented, which improves the basis of test vectors used to create the coarse grid matrices through a singular value decomposition and subsequent truncation of the basis. In Section 2, the discretization for Wilson fermions and their multigrid setup method are reviewed. In Section 3, we introduce the singular value decomposition setup method, the modifications for Wilson type fermions and the *streaming* method used to address the storage cost of the enlarged basis of test vectors. Section 4 demonstrates the applicability of the method on two separate gauge configurations from the MILC Collaboration and the Hadron Spectrum Collaboration and examines the volume dependence, as well as the performance of the streaming method. In Section 5, we summarize our results and provide future avenues of research.

2. Background

2.1. Multigrid for Wilson type fermions

Classical multigrid fails due to the near random nature of the $SU(3)$ gauge links. Only with the advent of adaptive multigrid was it possible to obtain a preconditioner for the Wilson-Dirac operator that significantly mitigated critical slowing down of the solver, that is, the number of iterations was increasing more slowly as the operator mass approached the critical mass. Under adaptive multigrid, the smoother is tested on the homogeneous equation $D\psi_i \approx 0$ to create a set of k near null or test vectors, $\Psi = [\psi_1, \dots, \psi_k]$, with random initial guesses, ϕ_i . That is equivalent to do

$$\psi_i = \phi_i + \psi'_i \quad \text{where } D\psi'_i \approx -D\phi_i, \quad \text{for } i = 1, \dots, k. \quad (2)$$

This procedure generates ψ_i rich in components of the error that the smoother is not able to easily resolve, i.e. the low frequency components of the error. The test vectors generated display the phenomenon known as *local coherence*, wherein the smooth test vectors approximate the low lying eigenspace of D on a given domain [12,13]. Thus, in the smooth aggregation approach, these smooth test vectors are then blocked according to a domain decomposition of the underlying grid, which means that each test vector is split into multiple vectors, each having support only on one of the subdomains. These vectors then become the columns of the *prolongator* matrix, P . The *restriction* matrix, R , is typically taken to be $R = P^\dagger$. However, the success of the multigrid

preconditioner for the Wilson-Dirac operator relies on “chiral splitting”, wherein the smooth test vectors are further split into chiral components by applying the projectors $1 \pm \gamma_5$ which split the spin components of Ψ such that

$$\begin{aligned} \Psi^{(\alpha=0)} &= [\psi_1^{(\beta=1,2)}, \dots, \psi_k^{(\beta=1,2)}] \\ \Psi^{(\alpha=1)} &= [\psi_1^{(\beta=3,4)}, \dots, \psi_k^{(\beta=3,4)}], \end{aligned} \quad (3)$$

where α labels the chiral index and β labels the spin index of the test vectors. Color indices have been suppressed for brevity. The blocked and chirally-split test vectors are then orthonormalized on each domain of the spacetime lattice $\Lambda_j \in \Lambda = \{\Lambda_1, \dots, \Lambda_d\}$ and the prolongator is formed as

$$P_{aij}(x) = \begin{cases} \psi_i^{(\alpha)}(x) & \text{if } x \in \Lambda_j, \\ 0 & \text{otherwise} \end{cases}, \quad (4)$$

where x denotes the lattice coordinate. This results in a block diagonal matrix of dimension $n \times 2kd$, where $d = |\Lambda|$. The use of the projectors $1 \pm \gamma_5$ enforces the symmetry $\gamma_5 P = P \sigma_3$, where σ_3 is the third Pauli spin matrix, and ensures that if a right singular vector is in the range of P , a left singular vector is in the range of R . The coarse grid matrix is then formed via

$$D_{\ell+1} = R D_\ell P. \quad (5)$$

This process is repeated recursively beginning with the coarse grid operator $D_{\ell+1}$ to form successively coarser grids. The projector $(1 \pm \gamma_5^\ell)$ is used to perform the chiral splitting, where $\gamma_5^\ell = \sigma_3$ for $\ell > 0$. The multigrid setup process is summarized in Algorithm 1. Typical applications of multigrid in lattice QCD utilize the multigrid hierarchy in a K -cycle [14] as a preconditioner for a flexible outer solver such as FGMRES or GCR. For a more detailed discussion on the formulation of the multigrid preconditioner for Wilson fermions, we refer the reader to [3] and Section III A of Ref. [5].

Algorithm 1 The multigrid setup algorithm for Wilson fermions.

Input: $D_\ell, \Lambda, d, k, \ell_{\max}$

Output: $D_{\ell+1}, R, P$

```

1: for  $\ell = 0, \dots, \ell_{\max} - 1$  do
2:   for  $i = 1, \dots, k$  do
3:     Smooth on  $D_\ell \psi_i \approx 0$  with random initial guess
4:   end for
5:   Apply the projectors  $(1 \pm \gamma_5^\ell)$  to  $\Psi = [\psi_1, \dots, \psi_k]$ 
6:   for  $\alpha = 0, 1$  do
7:     for  $j = 1, \dots, d$  do
8:       Orthonormalize  $\Psi^{(\alpha)}(\Lambda_j)$ 
9:     end for
10:  end for
11:  Form  $P$  as in Eq. (4); take  $R = P^\dagger$ 
12:  Form  $D_{\ell+1} = R D_\ell P$ 
13: end for
```

3. Chiral SVD of the multigrid basis

As mentioned earlier, a larger number of test vectors captures better the near null space providing a more effective preconditioner in terms of multigrid iterations. However, the size of the coarse grid matrix ($2kd$) also increases and thus it is not beneficial to increase k beyond a certain value¹. This tradeoff in smooth aggregation behavior is also common in other fields, and various approaches have been proposed to improve the quality of the test vector basis [15–17].

¹ a value of k around 24 is often used for the first level, and between 32 and 96 are often used for the subsequent levels.

One such approach finds $m > k$ smooth test vectors and truncates them to their best rank- k approximation using the singular value decomposition. Using the rank- k approximation as the multigrid basis of smooth test vectors was given in [16] and was shown to increase the performance of the multigrid preconditioner for two-dimensional anisotropic heat diffusion equations. It was also used in Ref. [15] to form composite aggregate preconditioners in bootstrap AMG. We review the rank- k approximation method given in [16] and then specify the modifications needed for Wilson fermions in lattice QCD.

To simplify the presentation of the method, we briefly assume only one degree of freedom per site (that is, every lattice site is then described by a single number) on a lattice of volume V and a uniform domain decomposition of the lattice where each domain Λ_j contains n_d lattice sites. Given a set of $m > k$ smooth test vectors at level ℓ , $\Psi^{(\ell)} = [\psi_1, \dots, \psi_m]$, consider its singular value decomposition restricted to a particular domain Λ_j , $\Psi^{(\ell)}(\Lambda_j) = U \Sigma V^\dagger$. Let $U_k \in \mathbb{C}^{n_d \times k}$ be the first k left singular vectors, $\Sigma_k \in \mathbb{R}^{k \times k}$ the first k singular values and $V_k \in \mathbb{C}^{m \times k}$ the first k right singular vectors. Then, the best rank- k approximation to $\Psi^{(\ell)}(\Lambda_j)$ in terms of the 2-norm error is $U_k \Sigma_k V_k^\dagger \approx \Psi^{(\ell)}(\Lambda_j)$. Then we can use the rank- k left singular vector basis for this approximation to obtain the prolongator for the next level, i.e., setting $P_j = U_k$ for each domain.

The extension of this method to Wilson fermions in lattice QCD is straightforward. The lattice sites are described by twelve degrees of freedom: 4 from the spin components and 3 from the color components. Guided by the required chiral splitting in the regular set up method, the singular value decomposition is performed separately on the chiral components of the initial set of m smooth test vectors given by (3), resulting in six degrees of freedom per lattice site for each chiral index. Let $U^{(\alpha)} \Sigma^{(\alpha)} V^{(\alpha)\dagger} = \Psi^{(\alpha, \ell)}(\Lambda_j)$ be the singular value decomposition of $\Psi^{(\alpha, \ell)}(\Lambda_j)$. Then, for $\alpha = 0, 1$, the prolongator is formed as

$$P_{aij}(x) = \begin{cases} u_i^{(\alpha)}(x) & \text{if } x \in \Lambda_j \\ 0 & \text{otherwise} \end{cases}. \quad (6)$$

As the singular value decomposition creates a basis of m orthonormal left singular vectors for each chiral index, the orthonormalization of step 8 in Algorithm 1 is not needed. We summarize the full multigrid set up algorithm utilizing the chiral SVD (CSVD) in Algorithm 2.

It is apparent that the cost of creating the multigrid preconditioner using Algorithm 2 scales as $\mathcal{O}(m)$ in the number of linear systems to be smoothed and incurs also a much smaller $\mathcal{O}(nm^2)$ SVD cost. Thus, it is more expensive and requires more storage than Algorithm 1. However, most applications in lattice QCD require solving Eq. (1) many times. Spectroscopy, which uses the distillation method [18], requires Eq. (1) be solved with $\mathcal{O}(10^4)$ right hand sides. In such a situation, the increased cost of generating more vectors would be heavily amortized. In the following section, we address the cost of storage through a streaming SVD, or incremental SVD (iSVD) method.

3.1. Streaming CSVD of the basis

The storage cost of the CSVD method in the preceding section scales linearly with m for a fixed domain size. A problem arises when it is necessary to choose a large value of m to obtain optimal multigrid performance (as shown in Section 4), but the computational platform has memory-constrained resources, such as GPUs, and therefore not all test vectors can be stored to compute the SVD. To address this, we implement an iSVD [19] method, which incrementally calculates approximate singular triplets of a matrix that is not available in its entirety but is streamed in windows of columns.

To facilitate the discussion of iSVD, we define the matrix

$$\Phi_k^{(\alpha)}(\Lambda_j) = U_k^{(\alpha)} \Sigma_k^{(\alpha)}, \quad (7)$$

where $U_k^{(\alpha)}$ and $\Sigma_k^{(\alpha)}$ are the first k left singular vectors and singular values of $\Psi^{(\alpha)}(\Lambda_j)$. Let each stream be labeled by subscript s , the number of streams be represented by n_s , and the size of the initial basis $m = k + r$,

Algorithm 2 The multigrid setup algorithm for Wilson fermions with CSVD.

Input: $D_\ell, \Lambda, d, m, k, \ell_{\max}$

Output: $D_{\ell+1}, R, P$

Require: $m \geq k$

```

1: for  $\ell = 0, \dots, \ell_{\max} - 1$  do
2:   for  $i = 1, \dots, m$  do
3:     Smooth on  $D_\ell \psi_i \approx 0$  with random initial guess
4:   end for
5:   Apply the projectors  $(1 \pm \gamma_5^\ell)$  to  $\Psi = [\psi_1, \dots, \psi_m]$ 
6:   for  $\alpha = 0, 1$  do
7:     for  $j = 1, \dots, d$  do
8:        $[U_k^{(\alpha)}, \Sigma_k^{(\alpha)}, V_k^{(\alpha)}] = \text{rank-}k \text{ SVD}(\Psi^{(\alpha)}(\Lambda_j))$ 
9:     end for
10:  end for
11:  Form  $P$  as in Eq. (6); take  $R = P^\dagger$ 
12:  Form  $D_{\ell+1} = R D_\ell P$ 
13: end for

```

where r is the number of test vectors generated at each stream. At the first stream, iSVD does not have any prior rank- k approximation, so $\Phi_k^{(\alpha)}(\Lambda_j)_0 = \emptyset$. For each stream then, iSVD updates the approximation as

$$\Phi_k^{(\alpha)}(\Lambda_j)_s = \text{rank-}k \text{ SVD}([\Phi_k^{(\alpha)}(\Lambda_j)_{s-1}, \Psi^{(\alpha)}(\Lambda_j)_s]) \quad (8)$$

where $\Psi^{(\alpha)}(\Lambda_j)_s$ are the r newly generated test vectors during the stream s , which are appended to $\Phi_k^{(\alpha)}(\Lambda_j)_{s-1}$. In Eq. (8), the operation rank- k SVD refers to obtaining the k dominant singular triplets of the matrix. During each stream, the SVD is performed on Eq. (8) and the first k left singular vectors and singular values are kept as in Eq. (7).

Algorithm 3 The iCSVD algorithm.

Input: $D_\ell, \Lambda, d, m, k, r, n_s$ **Output:** U_k

Require: $m = k + r$. Initially $m = r$ and $\Phi_m^{(\alpha)}(\Lambda_j)_{-1} = \emptyset$.

```

1: for  $s = 0, \dots, n_s - 1$  do
2:   for  $i = 1, \dots, r$  do
3:     Smooth on  $D_\ell \psi_i \approx 0$  with random initial guess
4:   end for
5:   Apply the projectors  $(1 \pm \gamma_5^\ell)$  to  $\Psi_s = [\psi_1, \dots, \psi_r]$ 
6:   for  $\alpha = 0, 1$  do
7:      $\Phi_m^{(\alpha)}(\Lambda_j)_s = [\Phi_k^{(\alpha)}(\Lambda_j)_{s-1}, \Psi^{(\alpha)}(\Lambda_j)_s]$ 
8:     for  $j = 1, \dots, d$  do
9:        $[U_k^{(\alpha)}, \Sigma_k^{(\alpha)}, V_k^{(\alpha)}] = \text{rank-}k \text{ SVD}(\Phi_m^{(\alpha)}(\Lambda_j)_s)$ 
10:      if  $s == n_s - 1$  then
11:        return  $U_k$ 
12:      else
13:         $\Phi_k^{(\alpha)}(\Lambda_j)_s = U_k^{(\alpha)} \Sigma_k^{(\alpha)}$ 
14:      end if
15:    end for
16:  end for
17: end for

```

Algorithm 3 gives our implementation of the iSVD algorithm, which accounts for the chirality of the test vectors. We therefore refer to it as incremental chiral SVD (iCSVD). Streaming SVD does not compute the singular vectors as accurately as SVD on the full set of vectors. However, its accuracy is typically higher than what is needed to build multigrid prolongators. Moreover, iCSVD allows us to sample a larger number of test vectors incrementally, keeping only the first k left singular vectors and singular values for each stream, resulting in a total space complexity of $\mathcal{O}(nm)$. In contrast, the non-streaming version of this method would need to store $n_s m$ test vectors for a comparable sample, resulting in a total space complexity of $\mathcal{O}(nn_s m)$. The iCSVD then allows us to sample more test vectors without the increase in storage cost and should result in a more effective preconditioner. Additionally, the use of iCSVD

slightly reduces the total time complexity, which scales as $\mathcal{O}(nm_s m^2)$, compared to a single SVD on all vectors. For example, if the number of test vectors used for the streaming version is m , and the number of test vectors for the non-streaming version is $n_s m$, then an equal number of test vectors have been sampled between the two methods. However, the time complexities between the two methods will differ by a factor of n_s , leading to a speedup in the total time spent performing the SVD by a factor of $\mathcal{O}(n_s)$. Algorithm 4 summarizes the full multigrid setup algorithm with the iCSVD method.

Algorithm 4 The multigrid setup algorithm for Wilson fermions with the iCSVD method.

Input: $D_\ell, \Lambda, d, m, k, r, n_s, \ell_{\max}$

Output: $D_{\ell+1}, R, P$

Require: $m = k + r$

```

1: for  $\ell = 0, \dots, \ell_{\max} - 1$  do
2:    $U_k = \text{iCSVD}(D_\ell, \Lambda, d, m, k, r, n_s)$ ;
3:   Form  $P$  as in Eq. (6); take  $R = P^\dagger$ 
4:   Form  $D_{\ell+1} = R D_\ell P$ 
5: end for

```

4. Numerical experiments

The setup method with CSVD is tested against regular multigrid within the MG_PROTO [20] library for two different types of lattice configurations. The first, which we refer to as Ensemble A, corresponds to an anisotropic lattice of dimension $32^3 \times 256$ with a pion of mass $m_\pi \approx 239$ MeV from the Hadron Spectrum Collaboration [21,22]. The gauge links arising from these configurations are stout smeared [23]. See [24] for more information relating to these lattices, including scale and mass determination. The second, which we refer to as Configuration B, corresponds to an isotropic lattice of dimension $32^3 \times 64$ with a pion of mass $m_\pi \approx 220$ MeV from the MILC Collaboration [25,26]. In this Clover on HISQ action, the gauge links are smeared with one iteration of HYP smearing [27]. Ref. [28] contains more information on the determination of the relevant physical properties of this ensemble. All experiments were performed on eight nodes with Intel Xeon Gold 6130 processors having 2×16 cores. In all experiments, the results are averaged over ten right hand sides and FGMRES was used as both a smoother and solver for each level. Unless explicitly stated otherwise, 8 iterations of a post-smoother were utilized on levels $\ell = 0, 1$ and the linear systems on levels $\ell = 1, 2$ were solved to a residual tolerance of $\|r_\ell\|_2 = 0.03\|b_\ell\|_2$. In all experiments, the first coarse level was created with an aggregate size of 4^4 , and subsequent levels were created using aggregates of size 2^4 . When comparing to the regular MG setup method, we use the same number of test vectors to ensure that the coarse grid matrices are the same size and density between the two methods.

4.1. Size of the multigrid basis

As the size of the initial multigrid basis m is increased, it is natural to expect that more information about the near null space is captured by the preconditioner, leading to an increase in convergence of the linear equations. Typical applications of multigrid in lattice QCD use between 24 and 32 test vectors. However, the CSVD allows us to calculate an initial basis size of $m \geq k$ and incorporate information from the test vectors that the regular setup method ignores. It is thus important to quantify the performance increase offered by increasing m .

Fig. 1 shows the mean execution time for solving the systems of linear equations for both Configuration A and Configuration B. The basis is truncated at $k = 24, 32$ for levels $\ell = 0, 1$, respectively. A general trend is observed for both configurations: as m increases, the mean execution time decreases. This is in line with expectations that additional information from the exposed smooth modes is made available to the preconditioner. It is also observed that a saturation in the speedup begins

between $m = 96$ and $m = 128$, indicating that beyond $m = 96$ the rate of new near-null information in test vectors reduces and the quality of the prolongators stabilizes.

4.2. Optimal truncation

The number of test vectors used to create the prolongation and restriction matrices is an important determination. Too few vectors results in a poor preconditioner and too many test vectors results in large and dense coarse grid matrices that are expensive to apply. It is thus important to examine the rank of the truncation utilized.

The singular spectra of the test vectors provides some information for this determination. Fig. 2 shows the singular spectrum of the chirally split test vectors for the first domain, which is representative of other domains. It is observed that the spectrum for Configuration A displays superlinear decay for approximately the first 16 singular values, and the spectrum for Configuration B displays superlinear decay for approximately the first 24 singular values for $\ell = 0$ for both chiral components. After these respective values, the spectrum displays linear decay. A similar observation is made for the level $\ell = 1$, with linear decay of the spectrum arising approximately after the first 30 singular values. The transition from superlinear decay to linear decay indicates that including left singular vectors into the basis beyond this approximate transition point will have a smaller overall contribution to the low rank approximation of the multigrid basis.

Fig. 3 displays the mean execution time of the system of linear equations when the truncation size, k , is varied for a constant value of $m = 96$ (as this m achieves most of the reduction in execution time). For both configurations, the optimal truncation is observed at $k = 24$. For Configuration A, this is comparable to $k = 16$, while Configuration B exhibits a distinct minimum at $k = 24$. We remark that the rank of the optimal truncation experimentally observed approximately agrees with the transition of the spectrum from superlinear to linear decay for both configurations. However, a detailed numerical study of this relation is beyond the scope of this study.

4.3. Optimal number of setup iterations

The number of set up iterations of the smoother to generate the test vectors may also have a large effect on the efficacy of the preconditioner. If not enough iterations of the smoother are used, some high frequency components of the error will remain. Conversely, too many iterations will begin to reduce the low frequency components of the error, resulting in a coarse grid matrix that does not capture those parts of the error. It is thus beneficial to examine the performance of the preconditioner while varying the number of setup iterations while generating the test vectors.

Figs. 4 and 5, for configurations A and B, respectively, display the total number of iterations required at each level ℓ as well as the mean execution time as the number of setup iterations vary. All levels perform the same number of setup iterations and, as previously, $m = 96$ and $k = 24$ are used. For both configurations, the total cost of solving the linear system is reduced compared to regular multigrid for all numbers of setup iterations. Additionally, the use of the CSVD results in a preconditioner that is less sensitive to the number of setup iterations.

Besides the lower total cost of solving the linear system, it is important to quantify the performance of the preconditioner at each level per iteration due to the use of the K -cycle. In a K -cycle, multiple iterations of the solver at level $\ell + 1$ occur for each iteration at level ℓ . To examine the cost per iteration on each level, the number of iterations at level $\ell + 1$ is normalized by the number of iterations on level ℓ . Fig. 6 displays the normalized cost for levels $\ell = 1, 2$ on both configurations. It is observed for both configurations across all number of setup iterations that the cost per iteration at the coarsest level is greater for the preconditioner that utilizes the CSVD, indicating that the coarsest level matrix generated by the CSVD method is more ill-conditioned than that of the coarsest level matrix generated by the regular MG method. This

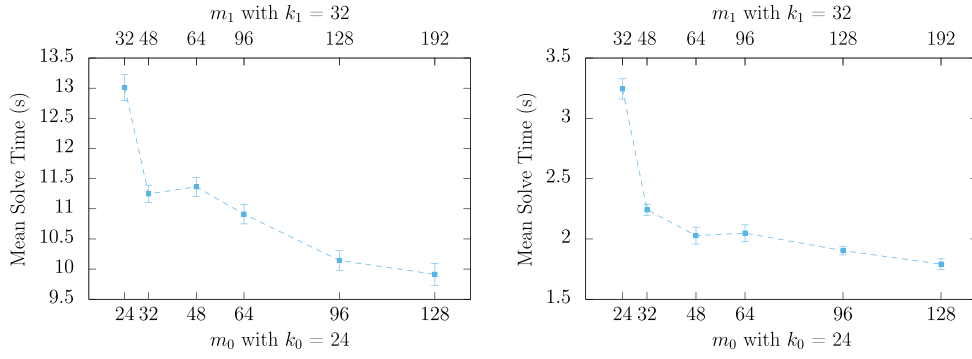


Fig. 1. The mean solve time as a function of the initial basis size for Configuration A (left) and Configuration B (right). The bottom and top y-axis displays m for levels $\ell = 0, 1$, respectively.

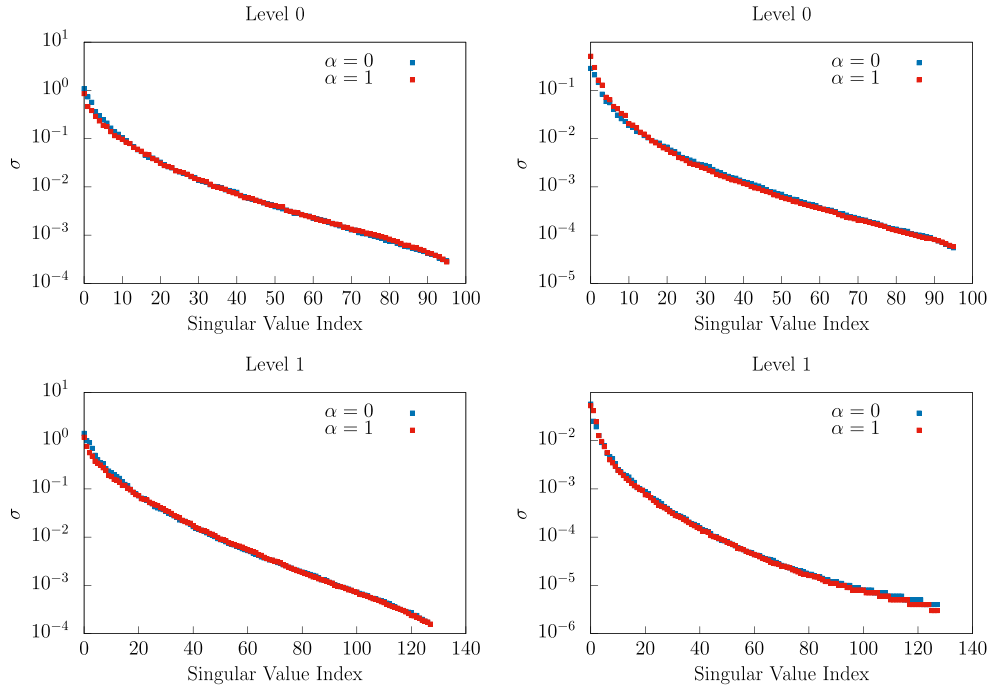


Fig. 2. The singular spectrum for the chirally split test vectors on the first domain for level $\ell = 0$ (top) and $\ell = 1$ (bottom) of Configuration A (left) and Configuration B (right).

is consistent with expectations that more information about the low frequency components of the error is being transferred to the coarse grids where more iterations can be performed at significantly lower cost.

4.4. Lattice volume scaling

We now examine the effectiveness of the CSVD method as the volume of the hypercubic lattice increases. Systems of linear equations are solved on three lattice volumes $L_s^3 \times L_t$, of spatial extent $L_s = 24, 32, 40$ and temporal extent $L_t = 64$, and $m_\pi \approx 220$ MeV with the Clover on HISQ action.² The parameters of the calculation were chosen from the near-optimal setup parameters and used across all three volumes.

To examine volume scaling, the mean execution time of the multigrid method is measured both with the CSVD method and without across the three volumes and reported in Fig. 7. For all three volumes, the preconditioner using CSVD shows a decrease in execution time. The speedup is found to be constant across all three volumes, indicating that the CSVD method maintains its effectiveness for larger volumes.

4.5. Streaming CSVD

We now turn to examining the effectiveness of the preconditioner generated with iCSVD method. For a direct comparison between the CSVD and iCSVD setup methods, we use a comparable number of test vectors between the two methods. The results of the previous sections showed an optimality with the use of $m = 96, 128$ initial test vectors, and a truncation to the first $k = 24, 32$ left singular vectors for both Configuration A and B. In the streaming version of the method, this corresponds to $m = 48, 64$ with $k = 24, 48$ and $r = 24, 32$ for $n_s = 4$ streams. The total number of sampled test vectors is then 96 and 128 on levels $\ell = 0$ and $\ell = 1$, respectively, for both methods.

Table 1 lists the mean number of iterations spent on each of the three multigrid levels during a linear solve for Configurations A and B, for the regular, CSVD and iCSVD setup methods. It is observed that there is no difference between the performance of the preconditioner created with CSVD and iCSVD methods, despite the iCSVD method producing singular vectors that are not as accurate.

In order to assess the performance of the preconditioner using iCSVD on a more difficult problem, we now examine the performance of the preconditioner when the bare quark mass is decreased to the ap-

² The same lattice of $L_s = 32$ is used here as in the preceding numerical results.

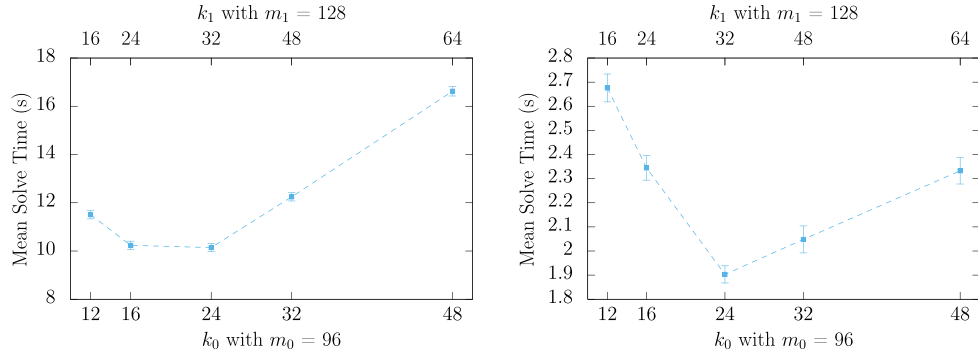


Fig. 3. The mean execution time for the system of linear equations when the degree of the truncation is varied using an initial basis size of $m = 96$ for Configuration A (left) and Configuration B (right). The bottom and top y-axis displays k for levels $\ell = 0, 1$, respectively.

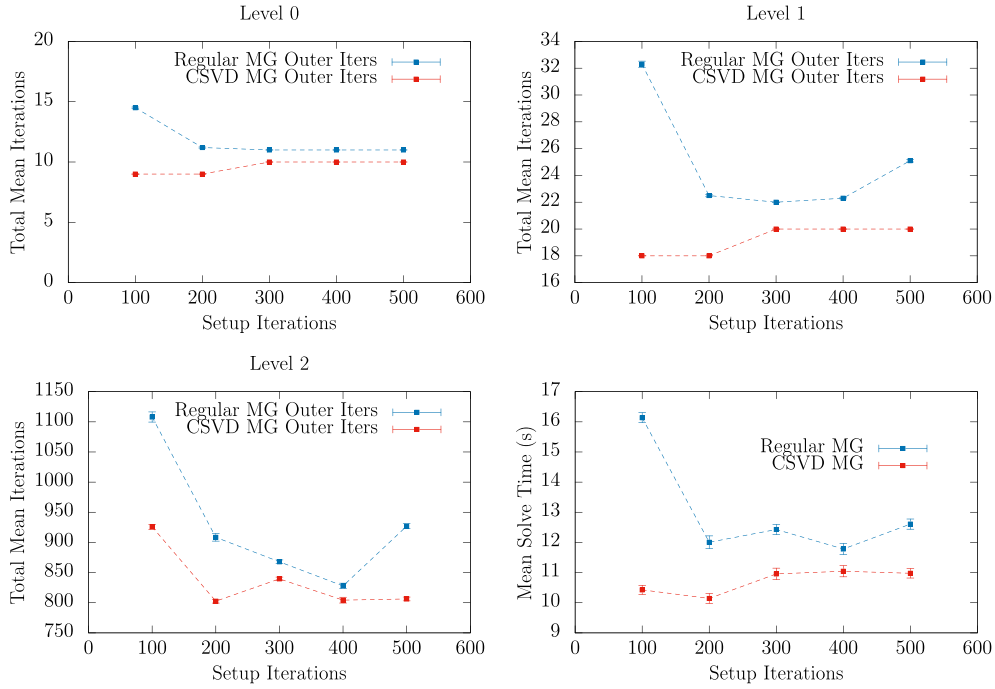


Fig. 4. The total number of iterations on $\ell = 0$ (upper left), $\ell = 1$ (upper right), $\ell = 2$ (lower left) and the mean solve time of the system of linear equations for Configuration A.

Table 1

The mean number of iterations spent on each multigrid level during a linear solve for all three setup methods with both Configuration A and B.

Level	Configuration A			Configuration B		
	Regular	CSVD	iCSVD	Regular	CSVD	iCSVD
$\ell = 0$	11.8	9.0	9.0	9.0	7.0	7.0
$\ell = 1$	23.6	18.0	18.0	22.7	14.0	14.0
$\ell = 2$	973.1	803.4	801.0	508.1	331.1	331.0

Table 2

The mean number of iterations spent on each multigrid level during a linear solve for the Regular, CSVD and iCSVD setup methods for Configuration B at $m_q \approx m_{crit}$.

Level	Regular	CSVD	iCSVD			
			$N_s = 4$	$N_s = 8$	$N_s = 12$	$N_s = 16$
$\ell = 0$	12.0	8.0	8.0	7.5	7.3	7.0
$\ell = 1$	34.0	16.0	16.0	15.0	14.6	14.0
$\ell = 2$	115.2	58.1	58.0	53.6	50.8	48.7
$\ell = 3$	635.5	533.1	525.1	515.8	458.1	458.0

proximate critical mass for Configuration B. While the computation of the quark propagator at critical mass is not relevant for most lattice QCD applications, it is appropriate for the evaluation of numerical algorithms as the condition number of the Wilson matrix drastically increases as the bare quark mass parameter approaches critical mass. The approximate critical mass was obtained through a fit of the pion propagator on 10 configurations from the ensemble of Configuration B, and was found to be $m_{crit} \approx -0.0804$. Consequently,

it was found that four multigrid levels are necessary as three levels did not provide the desired convergence for all methods. For the coarsest level, $m = 48$ with $k = 48$ and $r = 48$ were utilized. Adjusting the convergence criteria on level $\ell = 2$ to $||r|| = 0.06||b||$, as well as $||r|| = 0.6||b||$ for level $\ell = 3$ was also found to be necessary for the desired convergence. All other parameters remain the same as in previous experiments.

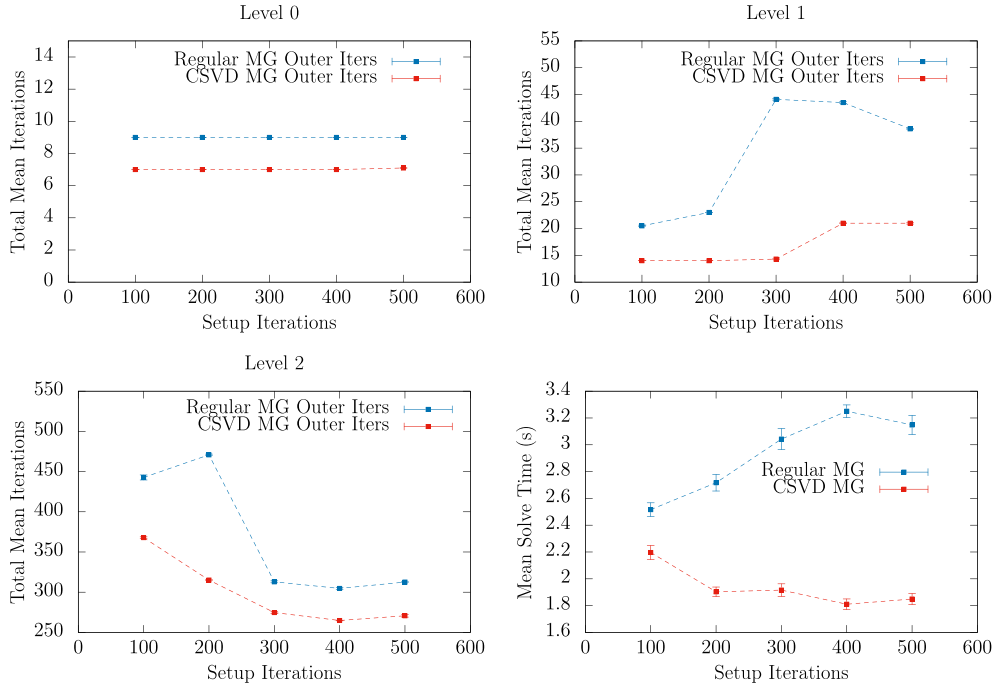


Fig. 5. As Fig. 4 for Configuration B.

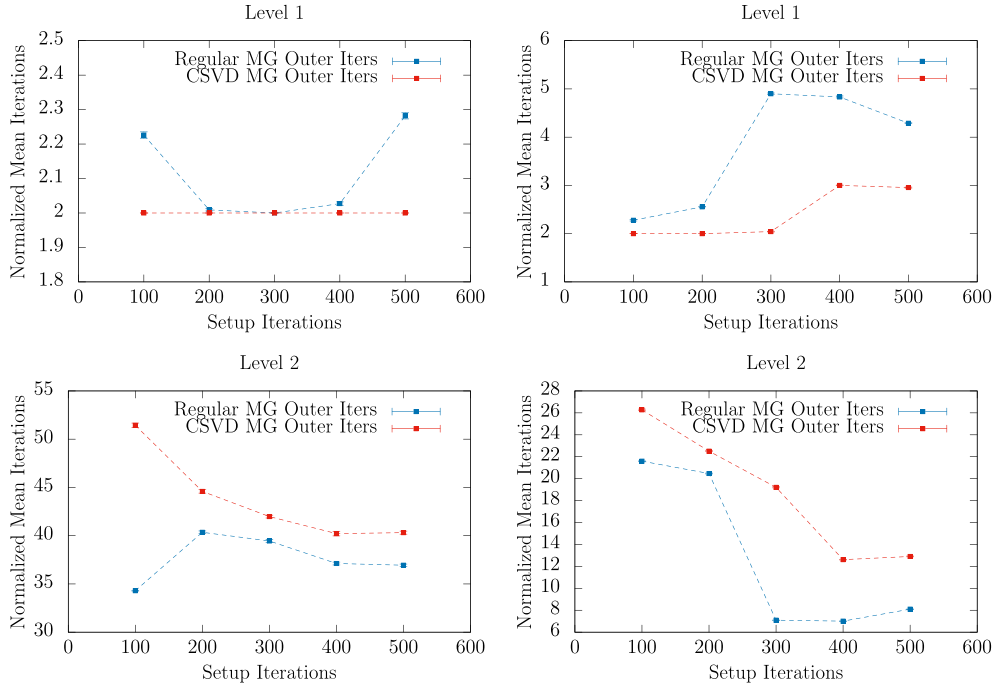
Fig. 6. The normalized number of iterations on level $\ell = 1$ (top) and $\ell = 2$ (bottom) for Configuration A (left) and Configuration B (right).

Table 2 displays the mean number of iterations for the regular and CSVD setup methods, as well as the iCSVD method for $n_s = 4, 8, 12, 16$. At this lower value of the quark mass, the number of outer iterations required in the linear solve has increased by 33% for the regular method (see Fig. 5). In contrast, the CSVD setup method only increases by approximately 15%. The benefits of the iCSVD are obvious at this lower value of the bare quark mass. At the largest number of streams utilized, the number of outer iterations is constant compared to the higher value of the bare quark mass utilized in Fig. 5. It also provides a speedup

factor of approximately 1.7 over the regular setup method at this value of the bare quark mass.

Table 3 shows the normalized mean iterations for each multigrid level performed during the solve of the linear equations. The qualitative behavior of the methods is analogous to that of Fig. 6. At $\ell = 1$, the number of iterations is reduced by approximately 40%. In contrast, $\ell = 2$ displays similar performance across all methods. The largest effect is seen on the coarsest level, where the number of normalized iterations increases drastically for the methods utilizing the

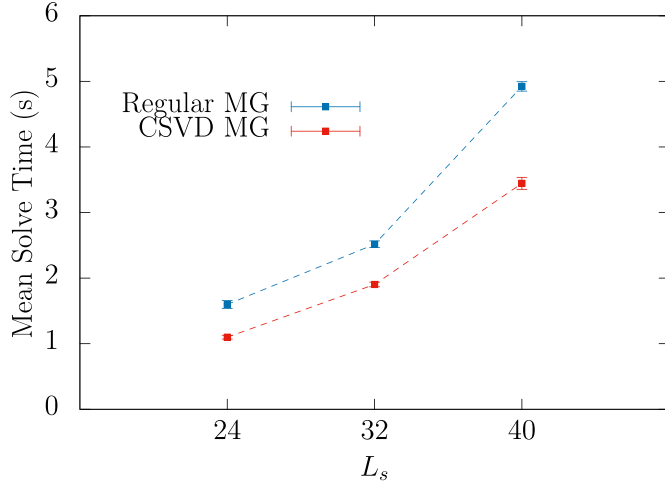


Fig. 7. The performance of both MG preconditioners as the lattice volume is increased.

Table 3

The mean normalized number of iterations spent on each multigrid level during a linear solve for the Regular, CSVD and iCSVD setup methods for Configuration B at $m_q \approx m_{crit}$.

Level	Regular	CSVD	iCSVD			
			$N_s = 4$	$N_s = 8$	$N_s = 12$	$N_s = 16$
$\ell = 1$	2.8	2.0	2.0	2.0	2.0	2.0
$\ell = 2$	3.4	3.6	3.6	3.6	3.5	3.5
$\ell = 3$	5.5	9.2	9.0	9.6	9.0	9.4

CSVD. This result again reinforces that the coarsest grid is capturing more components of the low lying part of the eigenspectrum, where the error corresponding to those components can be more efficiently reduced.

At the largest number of streams utilized, the setup phase of iCSVD is costly. However, as simulations reach physical quark mass and the lattice volumes become larger, the cost of such a setup phase may become necessary, as demonstrated by this artificially difficult problem.

5. Summary and future work

We have presented a modification to the setup algorithm for the multigrid preconditioner for systems of linear equations of Wilson fermions. The CSVD method utilizes a singular value decomposition of the chiral components of the test vectors restricted to a domain of the lattice. By calculating out a large basis of test vectors, the singular value decomposition is able to truncate the basis to the fewest vectors containing the largest contribution to the low rank approximation of the basis. In all numerical experiments, the use of the CSVD method results in a decrease in the time required to solve the system of linear equations. We also address the cost of storing the enlarged basis of vectors with iCSVD, which incrementally calculates the left singular vectors used in the prolongator. The performance of the multigrid preconditioner created using this method was shown to have the same performance as the non-streaming version. In numerical tests at critical mass, this method was shown to achieve a speedup factor of 1.7 in the number of fine grid iterations compared to the regular multigrid method.

A natural extension of the approach given in this study is the unification with the least squares interpolation approach [17], which has been used in a bootstrap AMG algorithm for the Wilson-Dirac operator in the two dimensional lattice Schwinger model [29]. In least squares interpolation, an optimal prolongator on the domain Λ_j is given by

$$P_j = V_f \Omega V_c^\dagger (V_c \Omega V_c^\dagger)^{-1} \quad (9)$$

where V_f are the fine grid vectors restricted to Λ_j and V_c are the coarse grid vectors. The diagonal matrix Ω contains the weights of the minimization, which are usually set to be the energy norm of the fine grid vectors. By setting $\Omega = I$, $V_f = U_k \Sigma_k V_k^\dagger$ and $V_c = \Sigma_k V_k^\dagger$, it can be shown that $P_j = U_k$, showing that the left singular vectors are the optimal prolongator for this choice of Ω . With this choice of Ω , the method described in this study can be viewed as a simplification of the multigrid eigensolver used in the bootstrap AMG method developed in Ref. [29], which is used to form approximate singular triplets of the matrix. In this sense, the CSVD and iCSVD methods compute an approximate multilevel SVD of the finest level matrix. However, this choice of Ω is not the only possible choice, and avenues in this direction will be explored in the future.

A further extension of this study could be realized through the use of an adaptive solver, as is employed in DD- α AMG [8] and described in Ref. [3]. In adaptive AMG, the multigrid hierarchy itself is used to expose components of the error that are not easily reduced by the smoother. This method comes with the added expense that the multigrid hierarchy must be rebuilt with the introduction of every new test vector and is therefore more expensive than smoothing on a predetermined number of test vectors. We remark that adaptive AMG and CSVD are not mutually exclusive. Indeed, the iCSVD method provides a natural framework for the combination of adaptive AMG and the CSVD methods. After creating the initial multigrid hierarchy through the CSVD method, the emerging solver could be used to generate additional test vectors for subsequent streams. This combination of the CSVD and adaptive AMG methods is therefore a promising future research direction.

CRediT authorship contribution statement

Travis Whyte: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Andreas Stathopoulos:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization; **Eloy Romero:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

TW acknowledges partial funding from the Exascale Computing Project (ECP), Project Number: 17-SC-20-SC, a collaborative effort of the U.S. Department of Energy, Office of Science and the National Nuclear Security Administration, as well as from a Royal Society Research Fellowship and the U.K. Science and Technology Facilities Council (STFC) [grant numbers ST/T000694/1, ST/X000664/1]. TW also acknowledges support from Science Foundation Ireland [grant number 21/FFP-P/10186] and Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of the CRC 1639 NuMerIQS - project no. 511713970. AS and ER acknowledge partial support by DOE SciDAC-5 grant (DE-FOA-0002589) and AS acknowledges partial support from National Science Foundation grant IIS-2008557. Part of this work was performed using computing resources at William & Mary with the software codes Chroma [30], QPhiX [31] and MG_PROTO [20]. We thank the MILC

Collaboration and the Hadron Spectrum Collaboration (www.hadspec.org) for making the gauge configurations available to us. The gauge configurations of the Hadron Spectrum Collaboration were generated using resources awarded from the U.S. Department of Energy INCITE program at Oak Ridge National Lab, the NSF Teragrid at the Texas Advanced Computer Center and the Pittsburgh Supercomputer Center, as well as at Jefferson Lab. We also thank Steven Gottlieb and Walter Wilcox for their help in obtaining the MILC configurations.

References

- [1] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, 2003.
- [2] T. Kalkreuter, Multigrid methods for propagators in lattice gauge theories, *J. Comput. Appl. Math.* 63 (1) (1995) 57–68. Proceedings of the International Symposium on Mathematical Modelling and Computational Methods Modelling 94. [https://doi.org/10.1016/0377-0427\(95\)00049-6](https://doi.org/10.1016/0377-0427(95)00049-6)
- [3] J. Brannick, R.C. Brower, M.A. Clark, J.C. Osborn, C. Rebbi, Adaptive multigrid algorithm for lattice QCD, *Phys. Rev. Lett.* 100 (2008) 041601. <https://doi.org/10.1103/PhysRevLett.100.041601>
- [4] M.A. Clark, B. Joó, A. Strelchenko, M. Cheng, A. Gambhir, R. Brower, Accelerating lattice QCD multigrid on GPUs using fine-grained parallelization, 2016. <https://arxiv.org/abs/1612.07873>. arXiv:1612.07873
- [5] R.C. Brower, E. Weinberg, M.A. Clark, A. Strelchenko, Multigrid algorithm for staggered lattice fermions, *Phys. Rev. D* 97 (2018) 114513. <https://doi.org/10.1103/PhysRevD.97.114513>
- [6] R.C. Brower, M.A. Clark, E. Weinberg, D. Howarth, Multigrid for chiral lattice fermions: domain wall, *Phys. Rev. D* 102 (2020) 094517. <https://doi.org/10.1103/PhysRevD.102.094517>
- [7] J. Brannick, A. Frommer, K. Kahl, B. Leder, M. Rottmann, A. Strebel, Multigrid preconditioning for the overlap operator in lattice QCD, *Numer. Math.* 132 (3) (2016) 463–490.
- [8] A. Frommer, K. Kahl, S. Krieg, B. Leder, M. Rottmann, Adaptive aggregation-based domain decomposition multigrid for the lattice Wilson–Dirac operator, *SIAM J. Sci. Comput.* 36 (4) (2014) A1581–A1608.
- [9] J. Espinoza-Valverde, A. Frommer, G. Ramirez-Hidalgo, M. Rottmann, Coarsest-level improvements in multigrid for lattice QCD on large-scale computers, *Comput. Phys. Commun.* 292 (2023) 108869. <https://doi.org/10.1016/j.cpc.2023.108869>
- [10] C. Alexandrou, S. Bacchio, J. Finkenrath, A. Frommer, K. Kahl, M. Rottmann, Adaptive aggregation-based domain decomposition multigrid for twisted mass fermions, *Phys. Rev. D* 94 (2016) 114509. <https://doi.org/10.1103/PhysRevD.94.114509>
- [11] C. Lehner, T. Wettig, Gauge-equivariant neural networks as preconditioners in lattice QCD, *Phys. Rev. D* 108 (2023) 034503. <https://doi.org/10.1103/PhysRevD.108.034503>
- [12] M. Lüscher, Deflation acceleration of lattice QCD simulations, *J. High Energy Phys.* 2007 (12) (2007) 011. <https://doi.org/10.1088/1126-6708/2007/12/011>
- [13] M. Lüscher, Local coherence and deflation of the low quark modes in lattice QCD, *J. High Energy Phys.* 2007 (07) (2007) 081. <https://doi.org/10.1088/1126-6708/2007/07/081>
- [14] Y. Notay, P.S. Vassilevski, Recursive Krylov-based multigrid cycles, *Numer. Linear Algebra Appl.* 15 (5) (2008) 473–487. <https://doi.org/10.1002/nla.542>
- [15] P. D’Ambra, P.S. Vassilevski, Improving solve time of aggregation-based adaptive AMG, *Numer. Linear Algebra Appl.* 26 (6) (2019) e2269.
- [16] E. Chow, An aggregation multilevel method using smooth error vectors, *SIAM J. Sci. Comput.* 27 (5) (2006) 1727–1741. <https://doi.org/10.1137/040608192>
- [17] A. Brandt, J. Brannick, K. Kahl, I. Livshits, Bootstrap AMG, *SIAM J. Sci. Comput.* 33 (2) (2011) 612–632. <https://doi.org/10.1137/090752973>
- [18] M. Peardon, J. Bulava, J. Foley, C. Morningstar, J. Dudek, R.G. Edwards, B. Joo, H.-W. Lin, D.G. Richards, K.J. Juge, et al., Novel quark-field creation operator construction for hadronic physics in lattice QCD, *Phys. Rev. D Part. Fields Gravitation Cosmol.* 80 (5) (2009) 054506.
- [19] M. Brand, Incremental singular value decomposition of uncertain data with missing values, in: *Computer Vision-ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part I* 7, Springer, 2002, pp. 707–720.
- [20] B. Joó, MG-PROTO: a multigrid library for QCD, (https://github.com/JeffersonLab/mg_proto/).
- [21] H.-W. Lin, S.D. Cohen, J. Dudek, R.G. Edwards, B. Joó, D.G. Richards, J. Bulava, J. Foley, C. Morningstar, E. Engelson, S. Wallace, K.J. Juge, N. Mathur, M.J. Peardon, S.M. Ryan, Hadron Spectrum Collaboration, First results from $2 + 1$ dynamical quark flavors on an anisotropic lattice: light-hadron spectroscopy and setting the strange-quark mass, *Phys. Rev. D* 79 (2009) 034502. <https://doi.org/10.1103/PhysRevD.79.034502>
- [22] R.G. Edwards, B. Joo, H.-W. Lin, Tuning for three flavors of anisotropic clover fermions with stout-link smearing, *Phys. Rev. D Part. Fields Gravitation Cosmol.* 78 (5) (2008) 054501.
- [23] C. Morningstar, M. Peardon, Analytic smearing of SU(3) link variables in lattice QCD, *Phys. Rev. D* 69 (5) (2004) 054501.
- [24] D.J. Wilson, R.A. Briceño, J.J. Dudek, R.G. Edwards, C.E. Thomas, for the Hadron Spectrum Collaboration, Quark-mass dependence of elastic πK scattering from QCD, *Phys. Rev. Lett.* 123 (2019) 042002. <https://doi.org/10.1103/PhysRevLett.123.042002>
- [25] A. Bazavov, C. Bernard, C. DeTar, W. Freeman, S. Gottlieb, U.M. Heller, J.E. Hetrick, J. Laiho, L. Levkova, M. Oktay, J. Osborn, R.L. Sugar, D. Toussaint, R.S. Van de Water, MILC Collaboration, Scaling studies of QCD with the dynamical highly improved staggered quark action, *Phys. Rev. D* 82 (2010) 074501. <https://doi.org/10.1103/PhysRevD.82.074501>
- [26] A. Bazavov, C. Bernard, J. Komijani, C. DeTar, L. Levkova, W. Freeman, S. Gottlieb, R. Zhou, U.M. Heller, J.E. Hetrick, et al., Lattice QCD ensembles with four flavors of highly improved staggered quarks, *Phys. Rev. D Part. Fields Gravitation Cosmol.* 87 (5) (2013) 054505.
- [27] A. Hasenfratz, F. Knechtli, Flavor symmetry and the static potential with hypercubic blocking, *Phys. Rev. D* 64 (2001) 034504. <https://doi.org/10.1103/PhysRevD.64.034504>
- [28] R. Gupta, Y.-C. Jang, B. Yoon, H.-W. Lin, V. Cirigliano, T. Bhattacharya, [Precision Neutron Decay Matrix Elements (PNDME) Collaboration], Isovector charges of the nucleon from $2 + 1 + 1$ -flavor lattice QCD, *Phys. Rev. D* 98 (2018) 034503. <https://doi.org/10.1103/PhysRevD.98.034503>
- [29] J. Brannick, K. Kahl, Bootstrap algebraic multigrid for the 2D Wilson–Dirac system, *SIAM J. Sci. Comput.* 36 (3) (2014) B321–B347. <https://doi.org/10.1137/130934660>
- [30] R.G. Edwards, B. Joo, The chroma software system for lattice QCD, *Nucl. Phys. Proc. Suppl.* 140 (2005) 832. [832(2004)]. arXiv:hep-lat/0409003 <https://doi.org/10.1016/j.nuclphysbps.2004.11.254>
- [31] B. Joó, D.D. Kalamkar, K. Vaidyanathan, M. Smelyanskiy, K. Pamnany, V.W. Lee, P. Dubey, W. Watson, Lattice QCD on Intel® Xeon Phi™ coprocessors, in: J.M. Kunkel, T. Ludwig, H.W. Meuer (Eds.), *Supercomputing*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 40–54.