

Bergische Universität Wuppertal

Fachbereich Mathematik und Naturwissenschaften

Institute of Mathematical Modelling, Analysis and Computational
Mathematics (IMACM)

Preprint BUW-IMACM 16/25

Costantia Alexandrou, Simone Bacchio, Jacob Finkenrath, Andreas
Frommer, Karsten Kahl, Matthias Rottmann

**Adaptive Aggregation-based Domain
Decomposition Multigrid for Twisted Mass
Fermions**

November 9, 2016

<http://www.math.uni-wuppertal.de>

less sensitive to the condition number and adapt them to the specific features of the discretization. This is particularly important for inversions using gauge configurations simulated with u- and d- quark masses fixed to their physical value, where improved solvers are indispensable for speeding up the computations and enabling the accumulation of enough statistics. Nowadays, such simulations are intensively being pursued by a number of lattice QCD collaborations and a lot of effort has been devoted in speeding up the solvers. A very successful approach has been based on multigrid ideas in preconditioning standard Krylov subspace solvers. There is a number of variant formulations of highly optimized multigrid solvers, which yield improvements of more than an order of magnitude in the case of the Wilson Dirac discretization, as reported in Refs. [2, 3, 4]. The most known implementations of multigrid algorithms, which are also available as open source software are the following: i) a two-level multigrid approach based on Lüscher’s inexact deflation [5] provided in the software openQCD [6]; ii) a multigrid approach developed in Refs. [7, 8, 9, 10], referred to as MG-GCR (Multi Grid - Generalized Conjugate Residual) which is part of the USQCD package QOPQDP [11] and; iii) an adaptive aggregation-based domain decomposition multigrid approach [1], referred to as DD- α AMG, recently made publicly available in the DDalphaAMG library [12].

Although these solvers have been developed for clover Wilson fermions they can be extended to other fermion discretization schemes as has been done for example for the overlap operator using DD- α AMG [13] or for Domain-Wall fermions using MG-GCR [14].

In this work we focus on the twisted mass (TM) Wilson Dirac operator. This discretization scheme has the advantage that all observables are automatically $\mathcal{O}(a)$ improved when tuned to maximal twist [15], where a is the lattice spacing. This formulation is thus particularly suitable for hadron structure studies, since the probe such as the axial current needs no further improvement in contrast to clover improved fermions. Furthermore, the presence of a finite twisted mass term bounds the spectrum of DD^\dagger from below by a positive term. This avoids exceptional configurations and, at the same time, gives an upper bound to the condition number, improving the convergence of numerical methods used in lattice QCD. Adding a small twisted mass term to the Wilson Dirac operator is also utilized as a ‘trick’ within simulations with the Wilson Dirac operator. For example, it is exploited in the case of quenched calculations using Wilson fermions [16, 17], in the simulation of clover improved Wilson fermions where it is combined with twisted mass reweighting [18, 19] or in mass reweighting [20]. Currently, the twisted mass formulation is the discretization scheme adopted by the European Twisted Mass (ETM) Collaboration for studying a wide range of observables. All these simulations employing twisted mass fermions will substantially benefit from an efficient multigrid method, especially for the analysis at the physical value of the light quark masses.

In this work we extend and optimize the DD- α AMG method to the case of the twisted mass Wilson Dirac operator. Therefore we exploit the Γ_5 -preservation within DD- α AMG which complements the twisted mass formulation. Our analysis shows that firstly the optimal parameters for the method are different from

those in the Wilson Dirac case, and secondly that, with appropriate parameter choices, we can gain speed-ups of more than one order of magnitude over methods previously used. The latter is similar to what has been observed in the case of clover improved fermions. Determining the appropriate parameter set is crucial since the tuning is a highly non-trivial task, and a complete analysis of how the method behaves for the different values of the parameters is still missing in the literature. We demonstrate in this study how the parameter space can effectively be reduced by tuning first the aggregation parameters. In doing so we also illustrate the robustness of the method. Moreover, our tuning approach can be easily applied to other aggregation-based multigrid solvers and fermion discretization schemes.

The paper is structured as follows: In Section 2 we introduce the twisted mass operator and discuss its properties with respect to multigrid preconditioning. In Section 3 we present our numerical results for twisted mass fermions, including the novel investigation of the aggregation parameters. Finally, in Appendix A we give an overview of the implementation of the method and how the DDalphaAMG library is extended to the case of twisted mass fermions [21].

2 DD- α AMG for twisted mass fermions

In this section we discuss the extension of the DD- α AMG solver to the case of the twisted mass fermion discretization scheme of the Dirac operator. For completeness we first review the twisted mass discretization scheme before we outline the various components of the multigrid preconditioner, pointing out the modifications necessary when going from the Wilson formulation to the twisted mass formulation, along the lines of Ref. [1].

We work on a four-dimensional hypercubic lattice defined by

$$\mathcal{V} = \{x = (x_0, x_1, x_2, x_3), 1 \leq x_0 \leq N_T, 1 \leq x_1, x_2, x_3 \leq L\} \quad (2)$$

with N_T the number of points in the temporal direction and L the number of points in the spatial directions x , y and z . Here and in what follows, the lattice spacing a is set to unity. The lattice volume is given by $V = N_T \cdot L^3$. Fermion fields are defined on the sites of the lattice, and each has four spin and three color degrees of freedom. The overall space is thus

$$\mathcal{V}_s = \mathcal{V} \times \mathcal{S} \times \mathcal{C} \quad (3)$$

where \mathcal{S} denotes the spin space and \mathcal{C} the color space.

2.1 Wilson Twisted Mass fermions

The Wilson Dirac operator $D_W = D(m)$ with the clover term can be written as

$$\begin{aligned}
 (D_W \psi)(x) &= \left((m+4)I_{12} - \frac{c_{sw}}{32} \sum_{\mu, \nu=0}^3 (\gamma_\mu \gamma_\nu) \otimes (Q_{\mu\nu}(x) - Q_{\nu\mu}(x)) \right) \psi(x) \\
 &\quad - \frac{1}{2} \sum_{\mu=0}^3 ((I_4 - \gamma_\mu) \otimes U_\mu(x)) \psi(x + \hat{\mu}) \\
 &\quad - \frac{1}{2} \sum_{\mu=0}^3 ((I_4 + \gamma_\mu) \otimes U_\mu^\dagger(x - \hat{\mu})) \psi(x - \hat{\mu}),
 \end{aligned}$$

with m the mass parameter and c_{sw} the parameter of the clover term. The gauge links $U_\mu(x)$ are $SU(3)$ matrices, and the set $\{U_\mu(x) : x \in \mathcal{L}, \mu = 0, 1, 2, 3\}$ is referred to as a gauge configuration. The γ -matrices act on the spin degrees of freedom of the spinor field $\psi(x)$ and fulfill the anti-commutation relation $\{\gamma_\mu, \gamma_\nu\} = 2 \cdot I_4 \delta_{\mu\nu}$ for $\mu, \nu = 0, 1, 2, 3$.

The Wilson Dirac operator satisfies the relation $\Gamma_5 D_W = D_W^\dagger \Gamma_5$ referred to as Γ_5 -hermiticity, where $\Gamma_5 = I_V \otimes \gamma_5 \otimes I_C$ acts on the space \mathcal{V}_s defined in Eq. (3) as a linear transformation of the spin degrees of freedom at each lattice site. In this paper we use a representation for $\gamma_5 = \gamma_0 \gamma_1 \gamma_2 \gamma_3$, which is diagonal in the spin space

$$\gamma_5 = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & -1 & \\ & & & -1 \end{pmatrix}. \quad (4)$$

The clover term, $Q_{\mu\nu}$, is given by

$$\begin{aligned}
 Q_{\mu\nu}(x) &= U_\mu(x) U_\nu(x + \hat{\mu}) U_\mu(x + \hat{\nu})^\dagger U_\nu(x)^\dagger + \\
 &\quad U_\nu(x) U_\mu(x - \hat{\mu} + \hat{\nu})^\dagger U_\nu(x - \hat{\mu})^\dagger U_\mu(x - \hat{\mu}) + \\
 &\quad U_\mu(x - \hat{\mu})^\dagger U_\nu(x - \hat{\mu} - \hat{\nu})^\dagger U_\mu(x - \hat{\mu} - \hat{\nu}) U_\nu(x - \hat{\nu}) + \\
 &\quad U_\nu(x - \hat{\nu})^\dagger U_\mu(x - \hat{\nu}) U_\nu(x - \hat{\nu} + \hat{\mu}) U_\mu(x)^\dagger.
 \end{aligned}$$

In case of the clover improved Wilson Dirac operator, the c_{sw} -term reduces the discretization error from $\mathcal{O}(a)$ to $\mathcal{O}(a^2)$ when c_{sw} is properly chosen.

The twisted mass formulation is a lattice regularization that allows automatic $\mathcal{O}(a)$ improvement by tuning only one parameter, namely the bare untwisted quark mass needs to be tuned to the so called critical mass. This formulation is particularly appropriate to hadron structure studies since the renormalization of local operators is significantly simplified with respect to the standard Wilson regularization. In the continuum, the twisted mass formulation is equivalent to the standard QCD action in a different basis. A mass term of the form $i\mu\tau_3 \otimes \Gamma_5$ can be added to the standard quark mass, where $\tau_3 = \text{diag}(1, -1)$ is the third Pauli matrix acting on a two-dimensional flavor space [15, 22]. By performing a suitable chiral transformation of the quark fields the mass term

can be rewritten in the standard form. The twisted mass theory becomes non-trivial on the lattice since Wilson fermions explicitly break chiral symmetry. We denote the twisted mass parameter by $\mu \in \mathbb{R}$ and apply the twist in the flavor space of the up (u-) and down (d-) quark. Then the twisted mass term acts with a positive shift given by $i\mu\Gamma_5$ on the u-quark operator and with a negative shift given by $-i\mu\Gamma_5$ on the d-quark operator. The twisted mass term breaks the isospin symmetry between the u- and the d-quark explicitly, which vanishes in the continuum limit. In the flavor space the operator applied to a spinor field is given as

$$\begin{bmatrix} D(\mu) & 0 \\ 0 & D(-\mu) \end{bmatrix} \begin{bmatrix} \psi_u \\ \psi_d \end{bmatrix} = \begin{bmatrix} (D_W(\psi_u)(x) + i\mu\Gamma_5\psi_u(x)) \\ (D_W(\psi_d)(x) - i\mu\Gamma_5\psi_d(x)) \end{bmatrix}, \quad (5)$$

where

$$D(\mu) = D_W + i\mu\Gamma_5 \quad (6)$$

is what we refer to as the twisted mass Wilson Dirac operator defined on the space \mathcal{V}_s . Adding a clover term reduces the isospin breaking induced by the twist.

Due to Γ_5 -hermiticity, the symmetrized Wilson Dirac operator $H_W = \Gamma_5 D_W$ is hermitian (and indefinite), such that we have

$$H_W = V\Lambda V^\dagger,$$

where the diagonal matrix Λ contains the eigenvalues λ_j of H_W (which are all real) and the unitary matrix V the corresponding eigenvectors. The ‘symmetrized’ twisted mass operator $H(\mu) = \Gamma_5 D(\mu) = H_W + i\mu$ thus satisfies

$$H_W + i\mu = V\Lambda V^\dagger + i\mu = V e^{i\Theta} \sqrt{\Lambda^2 + \mu^2} V^\dagger, \quad (7)$$

where the diagonal matrix $e^{i\Theta}$ contains the complex phases θ_j of the eigenvalues $\lambda_j + i\mu$ and $\sqrt{\Lambda^2 + \mu^2}$ their absolute values. For the non-symmetrized twisted mass operator $D(\mu)$, analogously to the Wilson case [1], this gives the singular value decomposition

$$D(\mu) = \Gamma_5 V e^{i\Theta} \sqrt{\Lambda^2 + \mu^2} V^\dagger = U \sqrt{\Lambda^2 + \mu^2} V^\dagger \quad (8)$$

with $U = \Gamma_5 V e^{i\Theta}$ and V being unitary. The smallest singular value $\sqrt{\lambda_i^2 + \mu^2}$ is thus not smaller than μ , which shows that a non-zero value of μ protects the twisted mass operator $D(\mu)$ from being singular, unlike the Wilson Dirac operator where this can happen for small quark masses m .

Similarly, for the squared twisted mass operator

$$D^\dagger(\mu)D(\mu) = (D^\dagger - i\mu\Gamma_5)(D + i\mu\Gamma_5) = D^\dagger D + \mu^2 = H_W^2 + \mu^2 \quad (9)$$

we have

$$H_W^2 + \mu^2 = V(\Lambda^2 + \mu^2)V^\dagger, \quad (10)$$

the eigenvalues of which are bounded from below by μ^2 .

Attaining automatic $\mathcal{O}(a)$ improvement on the lattice can be accomplished by tuning the PCAC (partial-conserved axial current) pion mass to zero. This corresponds to setting the angle Θ of the axial transformation $e^{i\Theta\tau_3\Gamma_5}$ to $\pi/2$. Then the renormalized light quark mass is directly proportional to the twisted mass parameter μ with $m_R = Z_P^{-1}\mu$ and Z_P the pseudoscalar renormalization constant. The breaking of the isospin symmetry in the twisted mass formulation results in the neutral pion being lighter than the charged pion. This slows down or even prohibits simulations for light quark masses close to the physical value. By adding the clover term to the action, the critical twisted mass value for the light quarks is significantly reduced compared to simulations without this term. This allows simulations at the physical point with a value of the lattice spacing around $a = 0.1$ fm or even larger [23].

A special property of the twisted mass operator is that at maximal twist the region just above μ^2 is densely populated with the eigenvalues of the squared operator, cf. Eq. (9). We illustrate this in Figure 1, which displays a histogram of the (scaled) ensemble averaged moduli of the eigenvalues of the non-squared symmetrized even-odd reduced (or preconditioned) twisted mass Dirac operator \hat{H} . The eigenvalues of the operator are measured on an ensemble simulated at a physical value of the light quark mass doublet, which we will refer to as the *physical ensemble*. \hat{H} is obtained using an even-odd ordering of the lattice sites such that

$$D = \begin{bmatrix} D_{oo} & D_{oe} \\ D_{eo} & D_{ee} \end{bmatrix}$$

where D_{oo} and D_{ee} are diagonal in spinor space. \hat{H} is then given as $\hat{H} = \Gamma_5 \hat{D}$ with

$$\hat{D} = (D_{ee} - D_{eo}D_{oo}^{-1}D_{oe}). \quad (11)$$

For $c_{sw} = 0$, i.e. without the clover term, the spectrum of \hat{D} is directly connected to the spectrum of the full operator D , e.g. in the case of the small eigenvalues we have

$$\frac{\lambda_D}{m+4} = 1 - \sqrt{1 - \lambda_{\hat{D}}/(m+4)} \approx \frac{1}{2} \frac{\lambda_{\hat{D}}}{m+4} \quad (12)$$

with λ_D and $\lambda_{\hat{D}}$ an eigenvalue of D and \hat{D} , respectively. Although this relation does not hold exactly for the hermitian even-odd reduced twisted mass Dirac operator, we found that numerically this relation still holds approximately for the eigenvalues close to μ . We find that the largest relative deviation of the smallest eigenvalue to the approximated cut-off 2μ is given by $|\lambda_{min} - 2\mu|/2\mu < 0.0005$ MeV. Thus we rescale the spectrum by a factor two and relate it to the energy scale of the $\overline{\text{MS}}$ -scheme defined at 2 GeV. The eigenvalue density is shown in Figure 1. In contrast to the spectrum of the Wilson Dirac operator shown in Refs. [5, 24], the density of the eigenvalues increases close to the physical quark mass.

Since $D(\mu)$ is non-normal, the left and right eigenvectors differ. For the twisted mass Dirac operator, the left eigenvectors of the u-quarks, represented

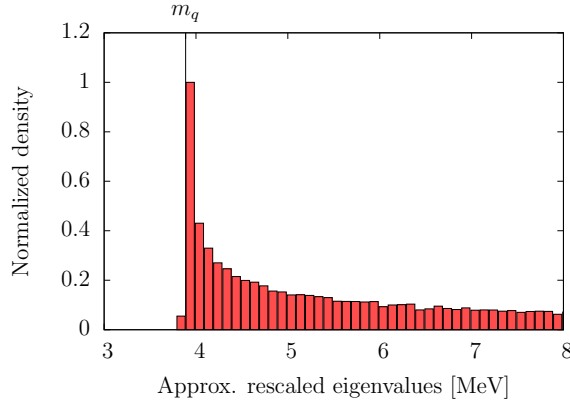


Figure 1: The density of the approximated rescaled eigenvalues of the hermitian even-odd reduced twisted mass Dirac operator measured on gauge configurations of the physical ensemble *cA2.09.48* (see Section 3). The quark mass is given by $m_q = 3.89$ MeV in the $\overline{\text{MS}}$ -scheme.

by $D(\mu)$, are connected to the right eigenvectors of the d-quarks, represented by $D(-\mu)$. If $\varphi_{j,L}^u$ and $\varphi_{j,R}^u$ are left and right eigenvectors of $D(\mu)$, respectively, with corresponding eigenvalue λ_j^u , then due to Γ_5 -hermiticity, we have

$$\begin{aligned} \lambda_j^u &= \varphi_{j,L}^u \dagger D(\mu) \varphi_{j,R}^u = \left(\varphi_{j,R}^u \dagger (D(\mu))^\dagger \varphi_{j,L}^u \right)^\dagger = \left(\varphi_{j,R}^u \dagger \Gamma_5 D(-\mu) \Gamma_5 \varphi_{j,L}^u \right)^\dagger \\ &= \left(\varphi_{j,L}^d \dagger D(-\mu) \varphi_{j,R}^d \right)^\dagger = \text{conj}(\lambda_j^d). \end{aligned} \quad (13)$$

Thus, eigenpairs of $D(-\mu)$ are connected to the eigenpairs of $D(\mu)$ by the transformations $\lambda_j^d = \text{conj}(\lambda_j^u)$, $\varphi_{j,L}^d = \Gamma_5 \varphi_{j,R}^u$ and $\varphi_{j,R}^d = \Gamma_5 \varphi_{j,L}^u$.

2.2 Multigrid preconditioning for Twisted Mass fermions

Our task is to solve the linear system

$$D(\mu) \psi = (D + i\mu\Gamma_5) \psi = b \quad (14)$$

for the Wilson twisted mass operator $D(\mu)$. The idea is to precondition a flexible iterative Krylov solver at every iteration step using a multigrid preconditioner.

Let ψ denote the current approximate solution to Eq. (14), then the corresponding error is given by $\epsilon = D^{-1}(\mu)b - \psi$, and the residual r satisfies

$$r = b - D(\mu)\psi = D(\mu)\epsilon. \quad (15)$$

In lattice QCD, preconditioners like the Schwarz Alternating Procedure (SAP [25]) efficiently reduce error components belonging to ultra-violet (UV)-modes, i.e. error components belonging to eigenvectors to large eigenvalues. Thus the error ϵ is then dominated by infrared (IR)-modes, i.e. eigenvectors corresponding to

small eigenvalues. For larger volumes, the increasing number of IR-modes slows down the preconditioned Krylov method.

Multigrid methods deal efficiently with both, IR- and UV-modes, independently of the volume size. Here we focus on the DD- α AMG preconditioner. A generic preconditioning step is described via its error propagation

$$\epsilon \leftarrow (I - MD)^k (I - PD_c^{-1}(\mu)RD) (I - MD)^j \epsilon. \quad (16)$$

Therein M denotes the smoother, P denotes the interpolation which maps from a coarser space with less degrees of freedom to \mathcal{V}_s , and R denotes the restriction, the counterpart of P , which maps from \mathcal{V}_s to the coarser space. Having P and R , the coarse grid operator $D_c(\mu)$, which can be seen as a coarse version of $D(\mu)$, is defined by the Galerkin condition

$$D_c(\mu) = RD(\mu)P. \quad (17)$$

Furthermore, $I - MD$ denotes the error propagator of the smoother. The powers j and k denote it's repeated application, i.e. we have j pre-smoothing and k post-smoothing steps. This error propagator is supposed to act on the UV-modes, it reduces the error components belonging to UV-modes. $I - PD_c^{-1}(\mu)RD$ denotes the error propagator of the coarse grid correction which acts complementary to the error propagator of the smoother. It reduces the error components belonging to IR-modes. As an iteration prescription, the coarse grid correction itself is given by

$$\psi \leftarrow \psi + PD_c^{-1}(\mu)Rr. \quad (18)$$

This two-level approach can be recursively expanded to a multi-level approach by using again a two-level approach for the inversion of $D_c(\mu)$. Then on each new level ℓ , transfer operators R_ℓ and P_ℓ are employed in the construction of the next level operator $D_{\ell+1}(\mu) = R_\ell D_\ell(\mu) P_\ell$ generalizing Eq. (17), and a smoother M_ℓ is assigned. DD- α AMG uses a K-cycle approach [26], which means that at each level, a multigrid preconditioned Krylov method is performed until a prescribed decrease of the residual is obtained. Our numerical tests showed that the inversion of the coarsest Dirac operator D_L can be evaluated at low accuracy, which reduces the computational effort. This is not the case for the inexact deflation approach from Ref. [5] as explained in Ref. [1].

2.3 Coarse grid

The coarse grid correction, given in Eq. (18), should approximately remove the error corresponding to the IR-mode components on the fine grid. An ideal, yet naive, choice would be to span the prolongation operator P by the right eigenvectors corresponding to small eigenvalues of $D(\mu)$ and the restriction operator R by the left ones, since then the coarse grid correction would entirely remove the (right) IR-modes present in the error. Such a choice is very similar to exact deflation of the small eigenvalues in that the coarse grid correction would solve directly the linear system of the IR-modes. In practice, the computational effort

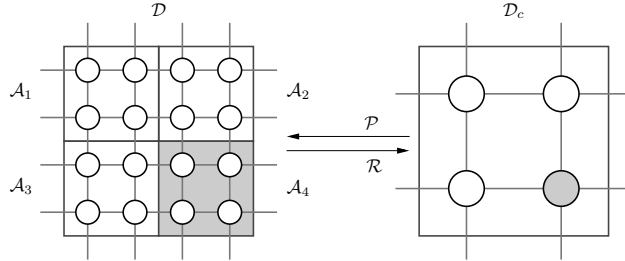


Figure 2: Block aggregation of the Dirac operator and symbolic representation of the movement from the fine operator to the coarse operator and vice-versa.

for generating such an “exact“ coarse grid projection is more than a magnitude higher than in aggregation based approaches, see e.g. Table 2 and 3 in Section 3.1.2, and it scales with $\mathcal{O}(V^2)$. Instead, the DD- α AMG solver uses a coarse grid correction, which is based on aggregation and the property of *local coherence* described in Ref. [5]. Local coherence implies that it is possible to approximate the subspace of the IR-modes by aggregates over just a small set of $\mathcal{O}(20)$ *test vectors* v_i . In this section, we discuss how to construct the aggregation based prolongation, the restriction and the coarse grid operators from the test vectors, while in Section 2.6 we will describe the strategy for generating the test vectors.

The coarse grid is obtained by mapping an aggregate \mathcal{A}_j of the fine grid to a single site of the coarse grid, where

$$\mathcal{A}_j = \mathcal{V}_j \times \mathcal{S}_j \times \mathcal{C}_j. \quad (19)$$

We will use static blocks with a fixed size given by

$$\mathcal{V}_j = \mathcal{T}_j \times \mathcal{Z}_j \times \mathcal{Y}_j \times \mathcal{X}_j \quad (20)$$

such that these blocks decompose the lattice \mathcal{V}_s as shown in Figure 2. The number of sites on the coarse lattice \mathcal{V}_c is then given by $N_b = V/V_b$ with V_b the block volume.

Now, a projection P and a restriction R between the fine grid and the coarse grid

$$\mathcal{V} \times \mathcal{S} \times \mathcal{C} \begin{array}{c} \xrightarrow{R} \\ \xleftarrow{P} \end{array} \mathcal{V}_c \times N_b, \quad (21)$$

can be constructed by using the test vectors v_i . For the prolongation P the test vectors are decomposed into blocks over the aggregates as

$$(v_1 | \dots | v_{N_b}) = \rightarrow P = \left(\begin{array}{c} \text{[Block 1]} \\ \text{[Block 2]} \\ \dots \\ \text{[Block } N_b] \end{array} \right) \begin{array}{c} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \vdots \\ \mathcal{A}_{N_b} \end{array}. \quad (22)$$

For the restriction operator, R^\dagger is constructed similarly from a possibly different set of test vectors w_i . For numerical reasons, the parts of all test vectors over a given aggregate are orthonormalized.

It is possible to extend the Γ_5 -hermiticity of the Wilson Dirac operator to the coarse grid operator. This can be done by decomposing the aggregation in the spin space \mathcal{S} with an aggregate

$$\mathcal{A}_{j,+} = \mathcal{V}_j \times \mathcal{S}_{0,1} \times \mathcal{C}, \quad (23)$$

which collects the two upper spin components 0 and 1, and an aggregate

$$\mathcal{A}_{j,-} = \mathcal{V}_j \times \mathcal{S}_{2,3} \times \mathcal{C}, \quad (24)$$

which collects the two lower spin components 2, 3. In the chiral basis, Γ_5 acts with $+1$ to the components of the aggregate $\mathcal{A}_{j,+}$ while with -1 on the other aggregates. Now, the coarse grid $\Gamma_{5,c}$ can be defined by $\Gamma_{5,c} = I_{\mathcal{V}_c} \otimes \tau_3 \otimes I_{N_v}$, where τ_3 acts on the different spin aggregates. This type of aggregation was proposed in Ref. [8] for the MG-GCR method and it is used as well in the DD- α AMG method, where it is termed *standard aggregation*. The Γ_5 -compatible prolongation P satisfies

$$\Gamma_5 P = P \Gamma_{5,c} \quad (25)$$

which can be represented as

$$\begin{pmatrix} I_{6 \cdot V_b} & & & \\ & -I_{6 \cdot V_b} & & \\ & & \ddots & \\ & & & \ddots \end{pmatrix} \begin{pmatrix} \text{[Diagram: 2x1 blocks]} \\ \text{[Diagram: 2x1 blocks]} \\ \ddots \\ \ddots \end{pmatrix} = \begin{pmatrix} \text{[Diagram: 2x1 blocks]} \\ \text{[Diagram: 2x1 blocks]} \\ \ddots \\ \ddots \end{pmatrix} \begin{pmatrix} I_{N_v} & & & \\ & -I_{N_v} & & \\ & & \ddots & \\ & & & \ddots \end{pmatrix}. \quad (26)$$

With the standard aggregation the coarse grid operator with a finite twisted mass term is given by

$$D_c(\mu) = R (D + i\mu\Gamma_5) P = R D P + i\mu\Gamma_{5,c} R P. \quad (27)$$

As pointed out in Ref. [1], the natural choice for the restriction operator is $R = (\Gamma_5 P)^\dagger$ since in this way the restriction approximates the subspace of the small *right* eigenvectors of the Wilson Dirac operator for the case that the interpolation approximates the small *left* eigenvectors. Using this formulation, the resulting coarse grid operator $R D(\mu) P$ is equivalent to $H_c + i\mu$ with $H_c = P^\dagger H P$, which is a complex shifted maximally indefinite operator (cf. Eq. (7)). Due to the commutativity relation between P and Γ_5 , given in Eq. (25), and already noted in Ref. [9], the coarse grid corrections obtained by $R = (\Gamma_5 P)^\dagger$ and $R = P^\dagger$ are identical. DD- α AMG thus uses $R = P^\dagger$, and the coarse grid twisted mass operator their is defined by

$$D_c(\mu) = P^\dagger D P + i\mu\Gamma_{5,c}. \quad (28)$$

$D_c(\mu)$ preserves several important properties of the fine grid operator $D(\mu)$. It preserves the sparse structure in that only neighboring aggregates are coupled. In the square coarse grid operator

$$D_c^\dagger(\mu)D_c(\mu) = D_c^\dagger D_c + \mu^2, \quad (29)$$

the eigenvalues are again bounded from below by μ^2 , and there is a $\Gamma_{5,c}$ -symmetry which reproduces the connection between the u- and d-quark operators as

$$D_c^\dagger(\mu) = P^\dagger(D + i\mu\Gamma_5)^\dagger P = P^\dagger(\Gamma_5 D \Gamma_5 - i\mu\Gamma_5)P = \Gamma_{5,c} D_c(-\mu)\Gamma_{5,c}, \quad (30)$$

Although the dimension of the coarse grid operator is reduced, it can develop a large number of small eigenvalues close to μ . This can critically slow down the convergence of a standard Krylov solver to be used on the coarsest grid such that the time spend in the coarsest operator inversions dominates by far the overall inversion time even though only poor accuracy is required. We therefore decrease the density of small eigenvalues of the coarsest grid operator by increasing the twisted mass parameter by a factor of δ given by

$$D_c(\mu, \delta) = D_c + i\delta\mu \cdot \Gamma_{5,c} \quad (31)$$

with $\delta \geq 1$ ¹.

We will analyze the effect of δ in detail in Section 3.3.2. As compared to the standard Wilson Dirac operator, for the twisted mass Dirac operator it turns out that the overall execution time is minimized if one relaxes even further the accuracy of the coarsest grid solve as will be described in more detail in Section 3.3.2 as well.

2.4 Smoother

In the DD- α AMG approach, a red-black Schwarz Alternating Procedure (SAP) is used as a smoother. This domain decomposition method was introduced to lattice QCD in Ref. [25], where it was used as a preconditioner. The lattice is partitioned into alternated “red(r)” and “black(b)” lattice blocks in a checkerboard manner, and the subdomains are obtained as the full color-spin space over the respective lattice block (see Eq. (20)). Re-ordering the Dirac twisted mass operator such that the red blocks come first, we obtain

$$D + i\mu\Gamma_5 = \begin{pmatrix} D_{rr} + i\mu\Gamma_5^r & D_{rb} \\ D_{br} & D_{bb} + i\mu\Gamma_5^b \end{pmatrix}, \quad (32)$$

where D_{rr} and D_{bb} are block diagonal matrices filled with the respective subdomains, while D_{rb} and D_{br} connect the neighboring blocks. Note that the operator D has only next neighbor interactions and thus blocks of a specific

¹This trick is used in heavy $N_f = 2$ twisted mass simulations at the charm quark mass and we thank Björn Leder for this suggestion.

color do not couple to the same color. The eigenmodes of the blocks have a higher cut-off than the full operator, given by $p_\nu = \pi/L_b$ in the free case for $\mu = 0$ due to the Dirichlet boundary conditions.

If now the role of the smoother is to reduce the UV-modes, a natural choice for the operator M in the error propagation (see (16)) is given by the inverse of the block operator, resulting in

$$E_{\text{SAP}} = (I - M_{\text{SAP}}D(\mu)) = (I - B_{rr}(\mu)D(\mu))(I - B_{bb}(\mu)D(\mu)), \quad (33)$$

where $B_{rr}(\mu)$ and $B_{bb}(\mu)$ are the block inverses defined as

$$B_{rr} = \begin{pmatrix} (D_{rr} + i\mu\Gamma_5^r)^{-1} & 0 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad B_{bb} = \begin{pmatrix} 0 & 0 \\ 0 & (D_{bb} + i\mu\Gamma_5^b)^{-1} \end{pmatrix}. \quad (34)$$

In practice, the (approximate) inversion of the blocks on the diagonal of B_{rr} and B_{bb} is performed with small computational cost by a few steps of an iterative method like the Minimal Residual method [1, 27]. Note that we fix the block size to coincide with the aggregates on each level of the multigrid hierarchy.

2.5 Krylov subspace methods

A Krylov subspace method preconditioned by the chosen multigrid approach has to be a flexible algorithm, since the smoother as well as the solver on the coarse grid system are non-stationary processes. Flexible solvers, which have been employed in multigrid preconditioning for lattice QCD are the flexible BiConjugate Gradient Stabilized method (BiCGStab, [28]), Generalized Conjugate Residual (GCR, [29]) and the Flexible Generalized Minimal RESidual ((F)GMRES, [30]) solver. In the DD- α AMG approach a FGMRES solver is used for the inversion of the fine grid operator and in the K-cycle for the inversion of the coarser operators except for the coarsest. The latter is inverted by even-odd preconditioned GMRES, i.e. GMRES is run for the even-odd reduced system $\hat{D}\phi_o = \eta_o - D_{oe}D_{ee}^{-1}\eta_e$ with \hat{D} from Eq. (11) and then ϕ_e in the solution $\begin{bmatrix} \phi_e \\ \phi_o \end{bmatrix}$ of $D\begin{bmatrix} \phi_e \\ \phi_o \end{bmatrix} = \begin{bmatrix} \eta_e \\ \eta_o \end{bmatrix}$ is retrieved as $\phi_e = D_{ee}^{-1}(\eta_e - D_{eo}\phi_e)$. Even-odd preconditioning is also used in the smoother when inverting the blocks. In both cases a speed-up of up to 50% compared to the full operator can be achieved. This can be explained by the fact that the small eigenvalues are increased by a factor of two as seen from Eq. (12).

In general, the accuracy of the coarse grid inversions may be very much relaxed as compared to the target accuracy of the fine grid inversion. Indeed, for DD- α AMG with the K-cycle strategy, optimal results are obtained when requiring the (approximate) inversions of the coarser operators to reduce the residual by just one order of magnitude.

2.6 Setup phase

In the setup phase we have to compute a sufficient number of approximate low modes v_i , which when chopped into aggregates, will approximate the IR-modes

well due to the local coherence property. We employ the setup algorithm from Ref. [1] for the twisted mass operator.

The vectors $\{v_i\}$ are generated by using a variant of block inverse iteration

$$v_i^{(k)} = D^{-1}(\mu)v_i^{(k-1)}, \quad i = 1, \dots, N_v, \quad k = 1, \dots, n_{\text{setup}}, \quad v_i^{(0)} \text{ chosen randomly,} \quad (35)$$

where the $v_i^{(k)}$ vectors converge to the eigenvectors with eigenvalues of smallest modulus. In practice, in order to maintain numerical stability, after each iteration k the vectors $v_i^{(k)}$ spanning the space of approximate IR-modes have to be orthonormalized, and $D^{-1}(\mu)$ is replaced by a multigrid iteration with prolongation operator constructed from the current set of approximate low modes $v_i^{(k)}$. For $k = 1$, where a multigrid hierarchy is not yet available, we just apply some steps of the SAP smoother. This approach results in a self-adapting procedure where the multigrid hierarchy is improved while using it to expose the small eigenmodes. Typically, a small number of setup iterations n_{setup} is sufficient.

3 Numerical results for twisted mass fermions

The DD- α AMG approach uses a wide range of parameters, and its time to solution will depend on a good choice of the parameters. The purpose of this section is therefore threefold. We first provide a set of default parameter choices, which in our extensive numerical testing turned out to yield good overall performance. Secondly, we show that for physically relevant configurations, appropriately chosen parameters yield speed-ups of about two orders of magnitude compared to standard methods for twisted mass fermions. We also show numerically that these parameters can be kept fixed over a whole statistical ensemble without a notable decrease in performance. Finally, we present an analysis on the dependence of several parameters which implies a general strategy for obtaining good parameters for a given ensemble and computer. All the presented numerical results have been obtained on SuperMUC phase 2 at the Leibniz Supercomputing Centre, an Intel Haswell Xeon computer on which we used up to 4096 cores. We have also performed runs on JURECA at the Julich Supercomputing Centre and on Cy-Tera at the Computation-based Science and Technology Research Center Cyprus obtaining compatible results. The stopping criterion of the overall iteration was fixed such that the residual is reduced by a factor of 10^9 .

3.1 Default parameters

Table 1 summarizes our default parameter set used for DD- α AMG. The parameter tuning was done for the ensemble *cA2.09.48* [23] with a lattice volume of $V = 96 \times 48^3$ and lattice spacing of $a = 0.0931(10)$ fm as determined from the nucleon mass. This ensemble was generated at a pion mass close to the physical one, namely $m_\pi = 0.131$ MeV [23].

	parameter		optimal
Multigrid	number of levels	n_ℓ	3
	number of setup iterations	n_{setup}	5
	number of test vectors on level 1	$N_{v,1}$	28
	number of test vectors on level 2	$N_{v,2}$	28
	size of lattice-blocks for aggregates on level 1	$V_{b,1}$	4^4
	size of lattice-blocks for aggregates on level ℓ , $\ell > 1$	$V_{b,2}$	2^4
Solver	mixed precision FGMRES		
	relative residual tolerance (restarting criterion)		10^{-6}
Smoother (SAP)	red-black multiplicative SAP		
	size of lattice-blocks on level 1		4^4
	size of lattice-blocks on level ℓ , $\ell > 1$		2^4
	number of post-smoothing steps		4
	MINRES iterations to invert the blocks		3
K-cycle	with single precision FGMRES		
	restart length		5
	number of maximal restarts		2
	relative residual tolerance (stopping criterion)		10^{-1}
Coarsest grid	solved by even-odd preconditioned GMRES		
	twisted mass parameter	μ_{coarse}	$5.2 \cdot \mu$
	restart length		100
	number of maximal restarts		5
	relative residual tolerance (stopping criterion)		10^{-1}

Table 1: The parameter set used in DD- α AMG, obtained by parameter tuning for the TM fermion ensemble *cA2.09.48* [23].

Solver	Setup time [core-hrs]	Inversion time [core-hrs]	Total iteration count of the fine grid solve	Total iteration count of coarse grid solvers
CG	–	174.8	26 937	–
CG-ede	1 527.4	5.4	649	–
DD- α AMG	13.3	0.9	16	2 988

Table 2: Results for clover Wilson fermions[†]. Comparison of CG (tmLQCD), eigCG (tmLQCD+ARPACK) with 800 eigenvectors and DD- α AMG with parameters from Ref. [4]. The results were for a $48^3 \times 64$ lattice from ensemble VII of Ref. [31] with $m_\pi = 0.1597(15)$ GeV.

3.1.1 Discussion

As explained in Section 2.6, in each iteration of the adaptive setup routine the currently available multigrid hierarchy is used to perform one iteration with the multigrid preconditioner on each test vector. If not stated otherwise, we used the parameters from Table 1 for the preconditioner in the setup phase and in the solve phase.

As the setup iteration proceeds the test vectors become more rich in low mode components. This also leads to a more ill-conditioned coarse operator D_c and higher iteration counts on the coarse grid (cf. Ref. [1]), i.e. one can observe a higher cost per setup iteration as the setup proceeds. This can be seen in in Table 3 where the setup times for $n_{\text{setup}} = 3$ and $n_{\text{setup}} = 5$ are stated. Indeed a factor of 5 in compute time between both setups can be observed. The suggested value $n_{\text{setup}} = 5$ from Table 1 thus yields a good value when several inversions ($\mathcal{O}(100)$ and more) with the same operator are desired. The relatively large setup time can be neglected in this case. We did not find that more setup iterations than $n_{\text{setup}} = 5$ yield substantial further improvement in the solve time. On the other hand, when solving for a few rhs, a good balance of setup and solve time has to be found. Therefore a smaller number of setup iterations like $n_{\text{setup}} = 3$ might be more suitable.

For larger pion masses one can, in principle, improve the generation of the subspace of small eigenmodes in the setup by using a smaller mass parameter m or a smaller twisted mass parameter μ , which helps to probe the small eigenmodes more rapidly within the setup phase. However, at least at the physical point, we do not find a significant improvement by using different mass parameters.

We use the same subspace for the u-quark with $+\mu$ and the d-quark with $-\mu$, i.e. we run the setup phase only once for both quarks. Although the eigenspace changes, numerically we do not find a large difference. This also saves computing time for many applications where the square operator has to

[†]In Tables 2 and 3 the timings for CG and eigCG have been normalized to 1.0 Gflop/s per core (average of standard performance ~ 0.7 Gflop/s and optimal performance ~ 1.3 Gflop/s); the rhs of the equation $D\psi = b$ has been randomly generated and in all the cases the propagator ψ has been computed to a relative precision of 10^{-9} .

Solver	Setup time [core-hrs]	Inversion time [core-hrs]	Total iteration count of the fine grid solve	Total iteration count of coarse grid solvers
CG	–	338.6	34 790	–
CG-eDe	6 941.1	9.8	695	–
DD- α AMG for $n_{\text{setup}} = 3$	7.7	2.5	28	16 619
DD- α AMG for $n_{\text{setup}} = 5$	38.3	1.5	15	11 574

Table 3: Results for TM fermions[†]. Comparison of CG (tmLQCD), eigCG (tmLQCD+ARPACK) with 1600 eigenvectors and DD- α AMG. The results were computed for the *cA2.09.48* ensemble [23] with $m_\pi = 0.131$ GeV.

be inverted, e.g. during the integration within the Hybrid Monte Carlo (HMC) algorithm [32, 33].

3.1.2 Comparison with CG

To put the improvements for twisted mass simulations in perspective, we start with an experiment for Wilson Dirac fermions, thus complementing the results from Ref. [4] for our target machine. We use the default parameter set from Ref. [4] and a configuration from ensemble *VII* [31] with a pion mass of $am_\pi = 0.05786$, a lattice spacing of $a = 0.0071$ fm and a lattice volume of $V = 64 \times 48^3$.

Table 2 presents a comparison of the inversion times of the Conjugate Gradient (CG) solver, the CG solver with exact deflation (CG-eDe) [27, 34] and the DD- α AMG solver. For CG and CG-eDe we use the publicly available software package *tmLQCD* [35] that is commonly used for simulations with the twisted mass operator and provides a count of floating point operations per second (*Flop/s*) for the CG solver. This makes it possible to rescale results obtained on different systems.

The conjugate gradient solver requires a positive definite hermitian matrix, which is obtained by solving the linear system with the squared even-odd reduced operator given by $\hat{D}^\dagger \hat{D}x = b'$ with $b' = \hat{D}^\dagger b$. This squares the condition number of the involved matrix.

The CG-eDe and the DD- α AMG solver involve a setup phase, which has to be done for each new configuration once before the linear system is solved. In the case of the CG-eDe solver, $\mathcal{O}(100)$ lowest eigenmodes of the squared even-odd reduced Dirac operator are calculated (here we use 800 for the case of Wilson case and 1600 for the twisted mass case). The number of eigenvectors in CG-eDe is obtained by optimizing the time to solution (setup + inversion time) for computing $\mathcal{O}(1000)$ rhs. Indeed the setup phase is extremely expensive which makes CG-eDe inefficient for a small number of rhs. The low mode computation is done by using the publicly available package ARPACK with tmLQCD [35].

Table 2 shows that in the case of Wilson Dirac fermions, the DD- α AMG solver speeds up the time to solution by roughly a factor of 200 compared to

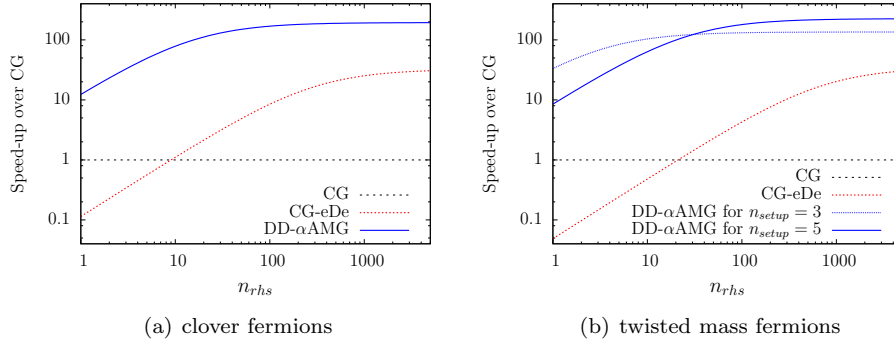


Figure 3: Speed-up over CG using the results of Table 2 and Table 3.

CG and roughly by a factor five compared to the CG-eDe solver. When including the setup time, DD- α AMG is roughly a factor of 12 faster than CG for one rhs, while CG-eDe is not competitive due to its computationally demanding setup.

In the twisted mass fermion case, we are able to achieve the same speed-ups as for the Wilson fermion case. This is not straight forward as it requires the coarsest grid twisted mass μ_{coarse} to be chosen different from the fine grid twisted mass μ . We choose $\mu_c = 5.2\mu$ (cf. Table 1) for reasons that will be explained in Section 3.3.2. The results for twisted mass fermions are shown in Table 3.

Indeed, we are able to achieve a speed-up in inversion time of roughly a factor of 220 compared to CG and roughly by a factor six compared to the CG-eDe solver, with the time for the setup being almost a factor of 100 less as compared to CG-eDe. For $n_{\text{setup}} = 3$ we used the parameters from Table 1 except for the number of test vectors on the fine level $N_{v,1}$ being 20 instead of 28. For $n_{\text{setup}} = 5$ we used a further optimized set which is given in the last row of Table 4. This yields another factor of 1.5 speed-up in the inversion time at the expense of increasing the setup time.

In Figure 3 we show the speed-up in total time (setup+solve) compared to one CG solve for DD- α AMG and CG-eDe as a function of the number of rhs n_{rhs} . The difference between the two blue curves in the figure on the right hand side is due to results from a different number of setup iterations (3 and 5) and a different number of test vectors (20 and 28), where 3 setup iterations are optimal for few rhs (blue dotted line) and 5 setup iterations for many rhs (blue line). In summary, the results for twisted mass fermions show that for one rhs DD- α AMG is roughly 30 times faster than CG for one rhs, 120 times faster for twenty rhs and 220 times faster for a thousand rhs.

3.2 Stability of optimal parameters

Empirically we find that the solver performance is stable for the tuned parameter set. Within the ensemble *cA2.09.48* we do not find any configurations where

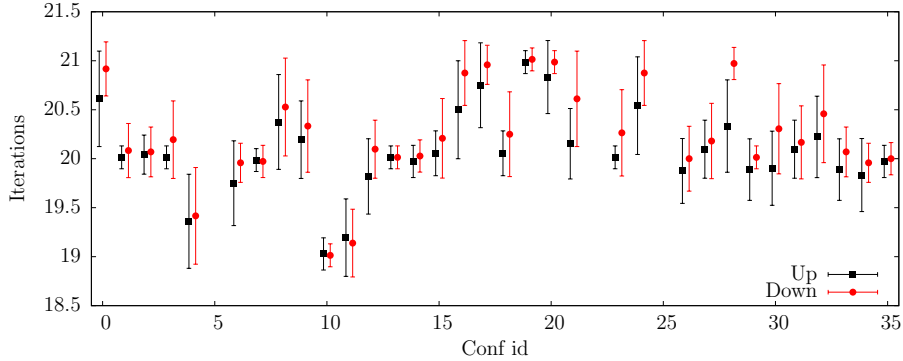


Figure 4: We depict the average iteration counts on computing quark propagators for several configuration of the ensemble $cA2.09.48$. For the black square points the setup is generated with the same μ while for the red circle points the setup is generated with a TM parameter μ with opposite sign.

the iteration counts or the time to solution differ by more than 5%, as can be seen in Figure 4. This behavior is also corroborated by the performance of the multigrid solver during the force computation in the HMC algorithm, where it shows very stable iteration counts for simulations at the physical point, see Figure 5.

3.3 Analysis of parameter settings and tuning strategy

As demonstrated in the previous section, well-tuned parameters are important for good performance, and they tend to be stable at least for configurations from a given ensemble. It is therefore advisable to invest some effort for obtaining good parameters. Since the parameter space is large, it cannot be searched exhaustively, and there is thus a need for a strategy how to tune the method in practice. Our suggestion for twisted mass simulations is to keep the default parameters given in Table 1, but tune the aggregation parameters and the twisted mass parameter on the coarsest level, μ_{coarse} . This is justified by the analysis that we explain in the subsequent sections as follows: In Section 3.3.1, we present a novel analysis of the aggregation parameters without tuning μ_{coarse} , i.e. we fix $\delta = 1$; in Section 3.3.2, we show the benefits obtained by increasing μ_{coarse} and we repeat the previous analysis; in Section 3.3.3, we demonstrate that also other multigrid approaches can benefit by an increasing μ_{coarse} . All the tests are performed on one configuration averaging the time to solution for the u- and d-propagator. This choice is motivated by the stability of the solver presented in the previous section.

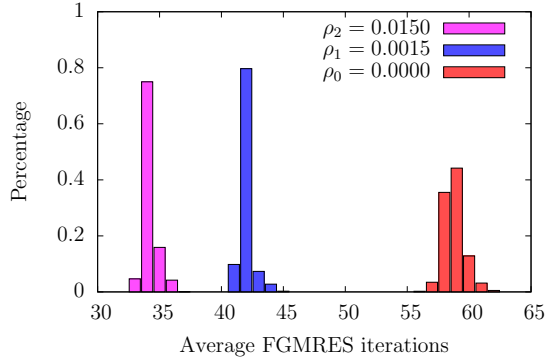


Figure 5: We show the FGMRES iteration counts averaged over the trajectory when the solver is used for computing the force terms in an HMC simulation. The data is for the ensemble *cA2.09.64* with statistics of 2000 trajectories and the squared operator $\hat{D}^\dagger \hat{D} + \rho_i^2$ is inverted, i.e. two inversions are performed with DD- α AMG inverting $\hat{D} + i\rho_i \hat{\Gamma}_5$ and $\hat{D}^\dagger - i\rho_i \hat{\Gamma}_5$ where $\hat{\Gamma}_5$ is the restriction of Γ_5 to the odd lattice sites. The parameter ρ_i sets the mass for the Hasenbusch preconditioning [36], the integration scheme is equivalent to the one presented in Ref. [23] for the ensemble *cA2z.09.48*. As a setup strategy within the HMC, we produce an initial set of test vectors using three setup iterations. Right before every inversion for ρ_0 we update the setup using one setup iteration at ρ_0 . The resulting preconditioner is used for all other Hasenbusch masses until the next update.

3.3.1 Aggregation parameters

Aggregation parameters are the number of the test vectors $N_{v,\ell}$ and the size of the lattice-blocks $V_{b,\ell}$ on each level ℓ . They should be tuned simultaneously since they define the size of the coarser Dirac operator and consequently the size of the projected subspace. In the present analysis, the solver is restricted to a 3-level implementation. We do not find an improvement in the time to solution by using a 4th level, which is also the result found in Ref. [4] for the Wilson operator with similar lattice sizes. On the other hand, the inversion time increases when just a 2-level multigrid method is used.

To optimize the aggregation parameters on the first level, we fix the values of the parameters on the second level for which we find a block size of $V_{b,2} = 2^4$ and a number of test vectors around 28 to work well. Fixing these parameters by setting $N_{v,2} = \max(28, N_{v,1})$ we analyze how the time needed to solve for one rhs and for the setup depends on the block size $V_{b,1}$ and the number of test vectors $N_{v,1}$. The results are depicted in Figure 6, for the *cA2.09.48* ensemble and a scaling parameter $\delta = 1$ on all levels has been used.

We find that every block size has an optimal number of test vectors as shown in Figures 6(a) and 6(b) that minimizes the time to solution. By fitting the data to a polynomial of order 2 in $N_{v,1}$ we estimate the minima and find that the optimal number of test vectors grows approximately linearly for block

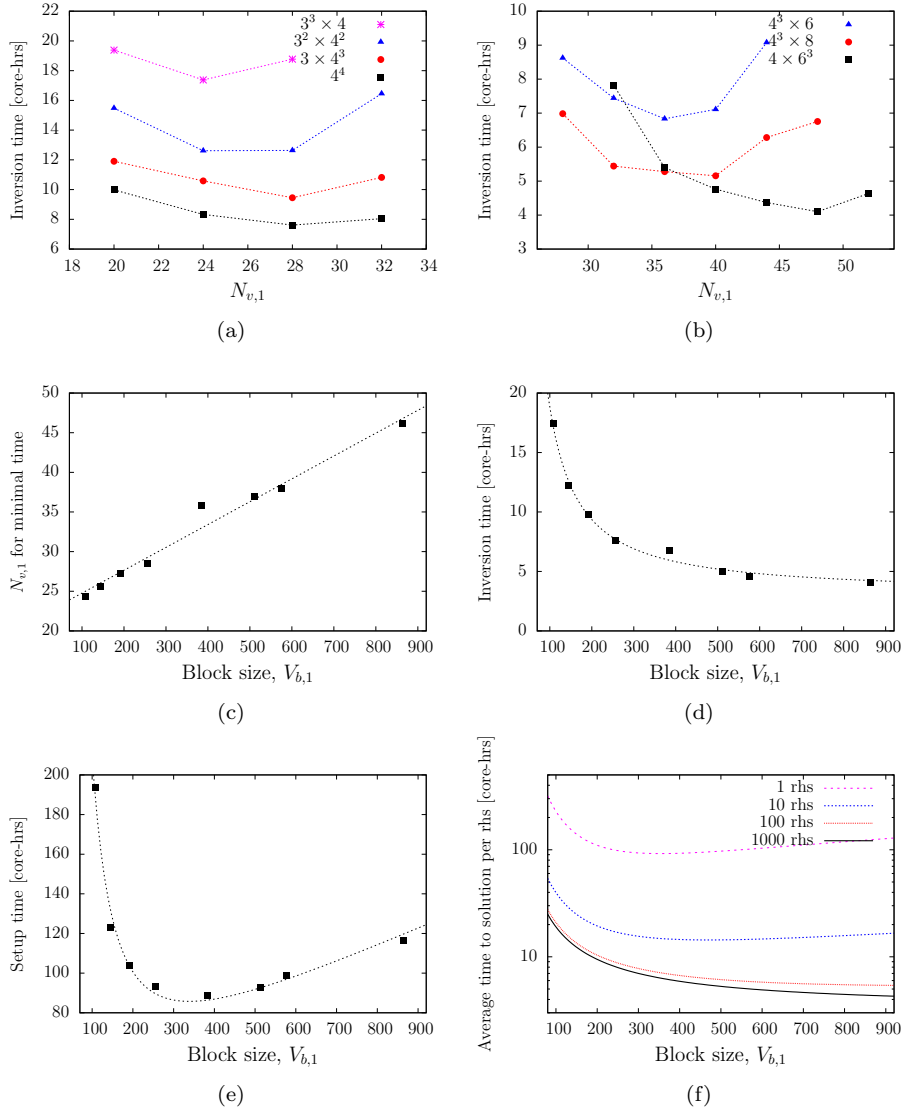


Figure 6: Analysis of the aggregation parameters for the *cA2.09.48* ensemble. For each set of raw data presented in (a) and (b) the position and value of the minimum has been extracted with a parabolic interpolation and displayed in (c) and (d), respectively. In (d), (e) and (f), $N_{v,1}$ and $V_{b,1}$ are connected according to the minima found in (c), i.e. (d) shows the inversion time and (e) the setup time, both for the minima from (c). The estimated average total time per right hand side $(t_{\text{setup}} + n_{\text{rhs}} t_{\text{solve}})/n_{\text{rhs}}$ is shown in (f). The fitting functions are explained in the text.

sizes $V_{b,1} < 8 \cdot 4^3$ with the block volume

$$N_{v,1} = \alpha + \beta V_{b,1}, \quad (36)$$

which is shown in Figure 6(c). This indicates a non-trivial connection between the fine grid size and the dimension of the coarse grid operator given by

$$\dim(D_{c,1}) = 2 \left(\frac{\alpha}{V_{b,1}} + \beta \right) V. \quad (37)$$

This implies that the optimal size of the coarse grid operator increases linearly with the volume V at fixed aggregation block size. For larger block sizes the behavior deviates from this linear dependence, however. For this case the minimum of the time to solution is already reached for a smaller number of test vectors. We can interpret this complex behavior as giving some insight into the non-trivial link between coarse grid size and local coherence.

The time to solution of the multigrid method is dominated by the solves with the coarse grid operator. We use a K-cycle and a 3-level approach where, due to fixing the blocks of the coarsest grid, it follows that $\dim(D_{c,1}) \propto \dim(D_{c,2})$. The numerical effort needed for a matrix-vector multiplication involving $D_{c,1}$ depends linearly on $\dim(D_{c,1})$. Allowing for an additional second order term we model the time for an inversion as

$$t_{solve} = \gamma' + \delta' \dim(D_{c,1}) + \varepsilon' \dim(D_{c,1})^2. \quad (38)$$

By using the dependence on $V_{b,1}$ of Eq. (37) we can rewrite t_{solve} with

$$t_{solve} = \gamma + \delta V_{b,1}^{-1} + \varepsilon V_{b,1}^{-2}. \quad (39)$$

Within this model, connecting $N_{v,1}$ and $V_{b,1}$ according to Eq. (36) and Figure 6(c), the time to solution can be fitted very well up to $V_{b,1} \approx 8^2 \times 4^2$ as shown in Figure 6(d). In Figure 7, we display the data obtained for the ensemble *cA2.09.64* with a lattice size of 128×64^3 . The other lattice parameters are the same as those for *cA2.09.48*. In contrast to the discussed analysis of *cA2.09.48*, the coarse grid scale factor is set to $\delta = 5$ on the coarsest grid. The full analysis on the dependence of the algorithm on δ is reserved for the next section. For both lattice sizes the inversion times reach the minimum for the same block volumes. The asymptotes γ from Eq. (39) are given by 1.64(13) for the *cA2.09.48* ensemble and by 6.3(5) for the *cA2.09.64* ensemble, which is an increase of $V^{5/4}$. As mentioned above, for larger block sizes, the linear connection of $N_{v,1}$ and $V_{b,1}$ does not hold and the solution time is increased. This is also observed for the time spent in the setup. We also observe that with increasing lattice volume the region where the linear dependence holds is shifted.

During the setup procedure the solver is applied on $N_{v,1}$ test vectors for several iterations. Here, the test vectors are orthonormalized at the end of each iteration and used for building the multigrid hierarchy. We model the setup time by allowing for a linear and quadratic term as

$$t_{setup} = \zeta + \eta N_{v,1} t_{solve} + \theta \dim(D_{c,1}), \quad (40)$$

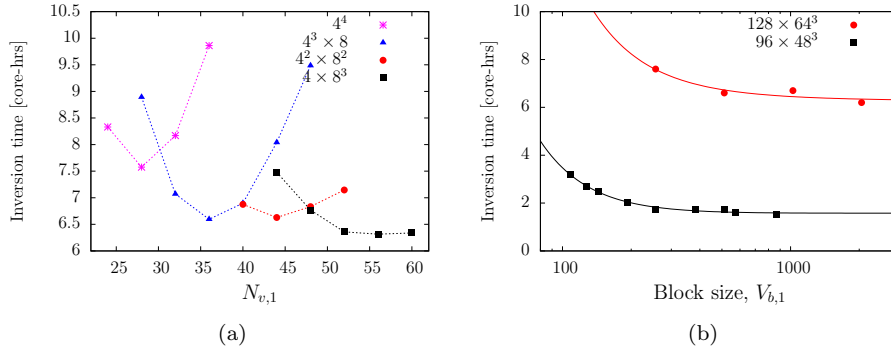


Figure 7: Analysis of aggregation parameters for $cA2.09.64$ and $\delta = 5$ on the coarsest grid. The sets of raw data are presented in (a); from each set, the value of the minimum has been extracted with a parabolic interpolation and plotted with red points in (b). The black points are for the ensemble $cA2.09.48$ and $\delta = 5$ on the coarsest grid.

where the amount of computation which does not involve the solver scales with $\dim(D_{c,1})$ at leading order. Figure 6(e) shows that the measured timings are indeed very well described by the Ansatz given in Eq. (40). We remark that for block sizes $V_b \sim 4^4$ up to $6^2 \times 4^2$ the time to solution shows a relatively large plateau, which makes the timings for the multigrid solver relatively stable. Further fine tuning in this region would lead only to small improvements.

The optimal choice for $N_{v,1}$ and $V_{b,1}$ depends on the number of rhs n_{rhs} . The total time is given by

$$t_{total} = t_{setup} + n_{rhs} t_{solve}, \quad (41)$$

and we find a non-trivial dependence of $V_{b,1}$ on n_{rhs} for an optimal time to solution, as depicted in Figure 6(f). This motivated our suggestion to consider two different values for n_{setup} , depending on the number of rhs.

3.3.2 Tuning the coarse grid scale factor δ

At physical quark masses, the density of the low-lying eigenvalues for the twisted mass operator increases compared to the Wilson Dirac operator, as explained in Section 2.1. Densely populated low eigenvalues slow down the iteration of the Krylov subspace solvers on the coarsest grid and thus of the whole multigrid method. This is much more pronounced for the twisted mass Dirac operator than it is for the Wilson Dirac operator. For the latter, Table 2 reports a total of around 3000 coarse grid iterations for the considered ensemble, whereas data depicted in Figure 8(a) show that for $\delta = 1$ we have roughly 40 times more coarse grid iterations in the twisted mass case, an unexpectedly large increase.

When using a larger twisted mass value on the coarsest grid operator, given by $\mu_{coarse} = \delta\mu$ with $\delta \geq 1$, we make the small eigenvalues less dense in the low-lying part of the spectrum. This speeds up the inversion time on the coarsest

Block size $V_{b,1}$	Test vectors $N_{v,1}$	δ for the coarsest μ $\mu_{\text{coarse}} = \delta\mu$	Setup time [core-hrs]	Inversion time [core-hrs]
$3^3 \times 4$	24	7.8	31.1	3.19(3)
2×4^3	24	6.6	27.5	2.71(5)
$3^2 \times 4^2$	24	8.6	23.5	2.47(2)
3×4^3	28	5.4	28.0	2.04(5)
4^4	28	5.2	22.2	1.75(4)
$4^3 \times 6$	36	4.5	37.4	1.74(6)
$4^3 \times 8$	40	4.0	40.7	1.73(5)
$4^2 \times 6^2$	40	4.1	37.7	1.59(6)
4×6^3	44	4.0	38.3	1.52(4)

Table 4: Summary of δ parameters yielding the optimal solve time for various block sizes $V_{b,1}$. The shown numbers were computed for the *cA2.09.48* ensemble and a relative residual tolerance of 10^{-9} . The number of test vectors was chosen as in Figure 6(c) according to the block size $V_{b,1}$. All other parameters were fixed to the values in Table 1.

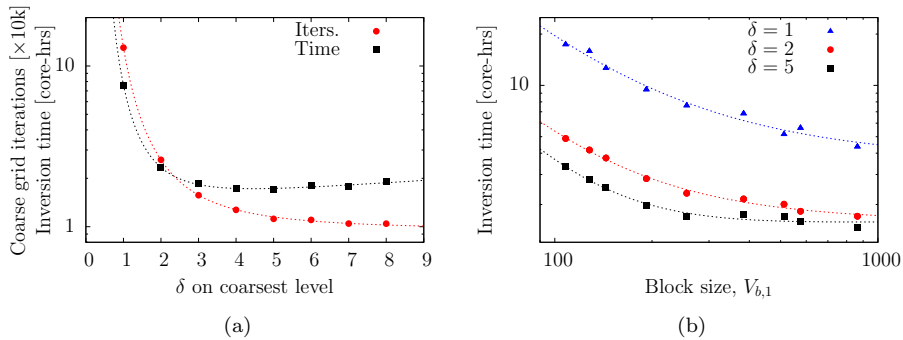


Figure 8: The number of coarse grid iterations as a function of the δ parameter and the inversion time for different δ -values depending on the block size. The results depicted in (a) are for a block size of $V_{b,1} = 4^4$ and compare the number of iterations on the coarse grid with the fine grid inversion time. The number of iterations has been scaled down by a factor 10 000. The behavior for the coarse grid iterations is proportional to $1/\delta^2$, while the inversion time has an additional term linear in δ due to the increase in the fine grid iteration count. The results depicted in (b) show the inversion time for different values of δ and different block sizes. The behavior is similar to the one observed in Figure 6(d).

grid. We also analyze the effect of the scaling factor δ as a function of the block size. The results are depicted in Figure 8 and we find that the optimal aggregation parameters determined in the previous section do not depend on the chosen coarse grid scale factor δ . In Table 4 we present the final results for the *cA2.09.48* ensemble, summarizing the δ parameters which minimize the inversion time for various block sizes and the corresponding optimal numbers of test vectors. By using a scaling factor $\delta = 2$ the iteration count on the coarsest grid is already reduced by a factor of five, which results into a speed-up of the fine grid inversion time by roughly a factor of three. Optimal performance is achieved for a relatively large plateau around $\delta \sim 5$. Obviously, a large scaling factor δ causes a distinct violation of the Galerkin condition, i.e. $P^\dagger D(\mu)P \neq D_c(\delta\mu)$. However, we only find a minor increase in the iteration count of the fine grid solver from 14 to 16 iterations.

3.3.3 The scale factor δ and different multigrid approaches

In addition to the studies of the delta parameter in DD- α AMG we tested the influence of a delta parameter in the QUDA implementation of MG-GCR [37] and in openQCD² [6]. The latter is restricted to a two-level approach. For both approaches we found that the iteration counts on the coarsest grid were reduced by using a scale factor $\delta > 1$, also resulting in reduced inversion and setup times. For $\delta \approx 5$ we observe speed-ups similar to those for DD- α AMG for the *cA2.09.48* ensemble.

For MG-GCR this behavior is to be expected since both approaches share the same type of Γ_5 -respecting interpolation (cf. Eq. (25)). However, the interpolation within the openQCD multigrid solver is not Γ_5 -respecting, resulting in a non-diagonal summand $P^\dagger \Gamma_5 P \delta\mu$ on the coarse grid instead of $\Gamma_{5,c} \delta\mu$ in Eq. (31).

We indeed find cases where the openQCD solver and DD- α AMG show different behaviors. When going to the *cA2.09.64* ensemble with a bigger volume, we observe that the optimal δ for the openQCD solver increases while it remains constant for DD- α AMG. Furthermore, when solving for $D(-\mu)$ with a setup built for $D(\mu)$ from the *cA2.09.48* ensemble, we find that the optimal δ increases by a factor of four within the openQCD solver whereas it remains constant for DD- α AMG.

For the Wilson fermion case it was reported in Ref. [4] that the two-level openQCD solver shows about the same performance as the three-level DD- α AMG approach. For the twisted mass ensemble *cA2.09.48*, the openQCD solver is roughly a factor of four slower than DD- α AMG.

²We use for the tests a modified openQCD-version, which is optimized for twisted mass fermions, i.e. where the even-odd reduced twisted mass Dirac operator is implemented.

4 Conclusions and outlook

The DD- α AMG solver is extended to the case of $N_f = 2$ twisted mass fermions. By carefully tuning the parameters we show that one can achieve speed-ups similar to those obtained in the case of Wilson fermions, to be precise, a factor of 200 compared to CG. This is mainly achieved by adapting the twisted mass parameter on the coarsest grid operator such that computing the coarsest grid correction becomes less time-consuming. Using a factor $\delta \sim 5$ to increase $\mu_{\text{coarse}} = \delta \cdot \mu$ on the coarsest level decreases the coarse grid iteration count by a roughly factor around 10 and improves the time to solution by a factor of around 4 for the case of ensembles generated at the physical pion mass.

An optimal set of parameters for different applications of the solver is presented and the strategy for tuning the aggregation parameters and the factor δ for the twisted mass parameter is discussed in detail. Using the optimal set of parameters, inversions are more than two orders of magnitude faster as compared to standard CG. Within the HMC, DD- α AMG achieves a speed-up of an order of magnitude compared to standard CG.

The DDalphaAMG library for twisted mass fermions is publicly available, and an interface to the tmLQCD software package is provided. The technical details are summarized in the Appendix A.

For the future we plan to extend multigrid methods to the heavy doublet sector of the twisted mass formulation.

Acknowledgments

This project has received funding from the Horizon 2020 research and innovation programme of the European Commission under the Marie Skłodowska-Curie grant agreement No 642069. S.B. is supported by this programme. We would like to thank Giannis Koutsou for fruitful discussions and for helping us accessing the configurations. We also thank Artur Strebel and Simon Heybrock for guidance with the DD- α AMG code and Björn Leder for his suggestion to shift the twisted mass on the coarse grid to speed-up the coarse grid solver. The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. for funding the project *pr74yo* by providing computing time on the GCS Supercomputer SuperMUC at Leibniz Supercomputing Centre, the computing time granted by the John von Neumann Institute for Computing (NIC) and provided on the supercomputer JURECA at Jülich Supercomputing Centre (JSC) through the grant *ecy00*, the High Performance Computing Center in Stuttgart for providing computation time on the High Performance Computing system Hazel Hen for calculating the eigenvalues through the grant *GCS-Nops (44066)*, and the computational resources on Cy-Tera at the Cyprus Supercomputing Centers through the grant *lspre258s1*.

A Overview of the DDalphaAMG library

The DDalphaAMG solver library, available at Ref. [12], has been recently released under GNU General Public License. This software package includes an implementation of the DD- α AMG for clover Wilson fermions as described in Ref. [1]. The implementation is of production code quality, it includes a hybrid MPI/openMP parallelization, state-of-the-art mixed precision and odd-even preconditioning approaches and also SSE3 optimizations. Implementation details can be found in Ref. [38].

Based on the DDalphaAMG code we developed a version, which supports twisted mass fermions, available at Ref. [21]. We added the following features to the library: $N_f = 2$ twisted mass fermions and twisted boundary conditions are supported, and different twisted mass shifts on the even and odd sites can be applied, which are required for the Hasenbusch mass preconditioning in the HMC when even-odd preconditioning is used.

An user documentation for the library can be found in `src/DDalphaAMG.h`, and a sample code that illustrates how to use the library interface functions can be found in `tests/DDalphaAMG_sample.c`. Moreover, the library has been integrated in the tmLQCD program [35], therefore we also adjusted the library interface. The interfaced code is available at Ref. [39]. Details on the employment of DDalphaAMG within tmLQCD are available at Ref. [40].

References

- [1] A. Frommer, K. Kahl, S. Krieg, B. Leder, and M. Rottmann. Adaptive aggregation based domain decomposition multigrid for the lattice Wilson Dirac operator. *SIAM J. Sci. Comput.*, 36:A1581–A1608, 2014.
- [2] M. Lüscher. Deflation acceleration of lattice QCD simulations. *JHEP*, 12:011, 2007.
- [3] J. C. Osborn, R. Babich, J. Brannick, R. C. Brower, M. A. Clark, S. D. Cohen, and C. Rebbi. Multigrid solver for clover fermions. *PoS, LATTICE2010:037*, 2010.
- [4] A. Frommer, K. Kahl, S. Krieg, B. Leder, and M. Rottmann. An adaptive aggregation based domain decomposition multilevel method for the lattice wilson dirac operator: multilevel results. 2013.
- [5] M. Lüscher. Local coherence and deflation of the low quark modes in lattice QCD. *JHEP*, 07:081, 2007.
- [6] M. Lüscher. OpenQCD software, version 1.2 and 1.4. Available at <http://luscher.web.cern.ch/luscher/openQCD>.
- [7] J. Brannick, R. C. Brower, M. A. Clark, J. C. Osborn, and C. Rebbi. Adaptive multigrid algorithm for lattice QCD. *Phys. Rev. Lett.*, 100:041601, 2008.

- [8] M. A. Clark, J. Brannick, R. C. Brower, S. F. McCormick, T. A. Manteuffel, J. C. Osborn, and C. Rebbi. The removal of critical slowing down. *PoS, LATTICE2008:035*, 2008.
- [9] R. Babich, J. Brannick, R. C. Brower, M. A. Clark, S. D. Cohen, J. C. Osborn, and C. Rebbi. The role of multigrid algorithms for LQCD. *PoS, LAT2009:031*, 2009.
- [10] R. Babich, J. Brannick, R. C. Brower, M. A. Clark, T. A. Manteuffel, S. F. McCormick, J. C. Osborn, and C. Rebbi. Adaptive multigrid algorithm for the lattice Wilson-Dirac operator. *Phys. Rev. Lett.*, 105:201602, 2010.
- [11] USQCD and J. C. Osborn. QCD Operations API on top of the QDP/C library (QOPQDP). Available at <http://usqcd-software.github.io/qopqdp>.
- [12] M. Rottmann, A. Strebel, S. Heybrock, S. Bacchio, and B. Leder. DDalphaAMG. Available at <https://github.com/DDalphaAMG/DDalphaAMG>.
- [13] J. Brannick, A. Frommer, K. Kahl, B. Leder, M. Rottmann, and A. Strebel. Multigrid preconditioning for the overlap operator in lattice QCD. *Numer. Math.*, 2015.
- [14] S. D. Cohen, R. C. Brower, M. A. Clark, and J. C. Osborn. Multigrid algorithms for Domain-Wall fermions. *PoS, LATTICE2011:030*, 2011.
- [15] R. Frezzotti and G. C. Rossi. Chirally improving Wilson fermions. 1. O(a) improvement. *JHEP*, 08:007, 2004.
- [16] K. Jansen, M. Papinutto, A. Shindler, C. Urbach, and I. Wetzorke. Quenched scaling of Wilson twisted mass fermions. *JHEP*, 09:071, 2005.
- [17] A. M. Abdel-Rehim, R. Lewis, and R. M. Woloshyn. Spectrum of quenched twisted mass lattice QCD at maximal twist. *Phys. Rev.*, D71:094505, 2005.
- [18] M. Lüscher and F. Palombi. Fluctuations and reweighting of the quark determinant on large lattices. *PoS, LATTICE2008:049*, 2008.
- [19] M. Lüscher and S. Schaefer. Lattice QCD with open boundary conditions and twisted-mass reweighting. *Comput. Phys. Commun.*, 184:519–528, 2013.
- [20] J. Finkenrath, F. Knechtli, and B. Leder. One flavor mass reweighting in lattice QCD. *Nucl. Phys.*, B877:441–456, 2013. [Erratum: Nucl. Phys.B880,574(2014)].
- [21] S. Bacchio and J. Finkenrath. DDalphaAMG, twisted mass version. Available at <https://github.com/sbacchio/DDalphaAMG>.
- [22] R. Frezzotti and G. C. Rossi. Chirally improving Wilson fermions. II. Four-quark operators. *JHEP*, 10:070, 2004.

- [23] A. Abdel-Rehim et al. Simulating QCD at the physical point with $N_f = 2$ Wilson twisted mass fermions at maximal twist. 2015.
- [24] L. Del Debbio, L. Giusti, M. Lüscher, R. Petronzio, and N. Tantalo. Stability of lattice QCD simulations and the thermodynamic limit. *JHEP*, 02:011, 2006.
- [25] M. Lüscher. Solution of the Dirac equation in lattice QCD using a domain decomposition method. *Comput. Phys. Commun.*, 156:209–220, 2004.
- [26] Y. Notay and P. S. Vassilevski. Recursive Krylov-based multigrid cycles. *Numerical Linear Algebra with Applications*, 15:473–487, 2008.
- [27] Y. Saad. Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems. 42(166):567–588, April 1984.
- [28] J. A. Vogel. Flexible BiCG and flexible Bi-CGSTAB for nonsymmetric linear systems. *Appl. Math. Comput.*, 188(1):226–233, 2007.
- [29] S. C. Eisenstat, H. C. Elman, and M. H. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.*, 20(2):345–357, 1983.
- [30] Y. Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.*, 14(2):461–469, 1993.
- [31] G. S. Bali, S. Collins, B. Glässle, M. Göckeler, J. Najjar, R. H. Rödl, A. Schäfer, R. W. Schiel, W. Söldner, and A. Sternbeck. Nucleon isovector couplings from $N_f = 2$ lattice QCD. *Phys. Rev.*, D91(5):054501, 2015.
- [32] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid monte carlo. *Phys. Lett.*, B195:216–222, 1987.
- [33] Steven A. Gottlieb, W. Liu, D. Toussaint, R. L. Renken, and R. L. Sugar. Hybrid molecular dynamics algorithms for the numerical simulation of quantum chromodynamics. *Phys. Rev.*, D35:2531–2542, 1987.
- [34] H. Neff, N. Eicker, T. Lippert, J. Negele, and K. Schilling. On the low fermionic eigenmode dominance in QCD on the lattice. *Phys. Rev.*, D64:114509, 2001.
- [35] K. Jansen and C. Urbach. tmLQCD: A program suite to simulate Wilson twisted mass lattice QCD. *Comput. Phys. Commun.*, 180:2717–2738, 2009.
- [36] M. Hasenbusch and K. Jansen. Speeding up lattice QCD simulations with clover improved Wilson fermions. *Nucl. Phys.*, B659:299–320, 2003.
- [37] R. Babich, K. Barros, R. Brower, M. Clark, J. Foley, J. Giedt, S. Gottlieb, B. Joó, C. Rebbi, G. Shi, and A. Strelchenko. QUDA: A library for QCD on GPUs QUDA, version 0.8. Available at <http://lattice.github.io/quda/>.

- [38] M. Rottmann. *Adaptive domain decomposition multigrid for lattice QCD*. PhD thesis, Bergische Universität Wuppertal, 2016.
- [39] S. Bacchio and J. Finkenrath. tmLQCD interfaced to DDalphaAMG. Available at <https://github.com/finkerath/tmLQCD/tree/DDalphaAMG>.
- [40] S. Bacchio, C. Alexandrou, J. Finkenrath, A. Frommer, K. Kahl, and M. Rottmann. DDalphaAMG for Twisted Mass Fermions. *PoS, LATTICE2016:259*, 2016.