Gauge-equivariant pooling layers for preconditioners in lattice QCD

C. Lehner[1, *] and T. Wettig[1]

[1] *Department of Physics, University of Regensburg, 93040 Regensburg, Germany*
(Dated: April 21, 2023)

We demonstrate that gauge-equivariant pooling and unpooling layers can perform as well as traditional restriction and prolongation layers in multigrid preconditioner models for lattice QCD. These layers introduce a gauge degree of freedom on the coarse grid, allowing for the use of explicitly gauge-equivariant layers on the coarse grid. We investigate the construction of coarse-grid gauge fields and study their efficiency in the preconditioner model. We show that a combined multigrid neural network using a Galerkin construction for the coarse-grid gauge field eliminates critical slowing down.

I. INTRODUCTION

Numerical simulations of quantum field theories such as Quantum Chromodynamics (QCD) continue to be our best systematically improvable method to obtain information on the nonperturbative features of the theory. These simulations are done on a finite space-time lattice on large supercomputers. In most cases, the run time of the simulations is dominated by the solution of the Dirac equation for a fixed gauge field. This is usually done by iterative algorithms whose iteration count is determined by the condition number of the matrix representing the linear system, in our case the Dirac operator. As we approach the interesting regions of parameter space, i.e., physical quark mass and continuum limit, the condition number of the Dirac matrix becomes very large, leading to critical slowing down. To deal with this problem, very sophisticated algorithms have been developed over the years. In particular, suitably constructed multigrid preconditioners have been shown to reduce or even eliminate critical slowing down [1–11]. Multigrid algorithms use restriction and prolongation operators to transfer fields from a fine to a coarse grid and back. In a recent paper [12] we discussed the construction of such multigrid preconditioners in the language of gauge-equivariant neural networks and showed that the multigrid paradigm can be learned efficiently by such networks. However, in Ref. [12] the restriction and prolongation layers were not learned but computed by hand. The aim of the present paper is to demonstrate that these two layers can also be learned by gauge-equivariant neural networks and perform as well as the traditional construction. We show that both the model of Ref. [12] as well as the new models discussed in the current work eliminate critical slowing down.

The construction of multigrid algorithms for lattice field theory has a long history, addressing both the Markov chain Monte Carlo sampling of fields and the computation of propagators. In the late 1980s and early 1990s a number of groups devised several multigrid schemes aimed at eliminating critical slowing down for different lattice field theories, gauge groups, and fermion discretizations [13–39]. There was even an early attempt to use neural networks in this context [40]. These works used gauge-equivariant constructions of restriction and prolongation operators to address high-frequency noise from the gauge degrees of freedom. Note that in these papers gauge equivariance is referred to as gauge covariance, as is common in quantum field theory. Another important guiding principle is the approximate preservation of the space spanned by the low eigenmodes of the Dirac operator on the coarse grid. The observation of "local coherence" of the low modes [2] implies that this space can be approximated locally by a relatively small number of suitable vectors. State-of-the-art multigrid algorithms make use of this observation in the construction of the restriction and prolongation operators. Our explicit construction of these operators in [12] was also based on this observation. Here, we replace this construction by gauge-equivariant pooling and unpooling layers but are still guided by the same objectives. These layers are parametrized by gauge-invariant spin matrices which are learned in the present work. In future work, we will construct models that, for a given gauge configuration, provide these matrices as output features.

There is a growing body of related work. Several authors have constructed multigrid algorithms, or elements thereof, using neural networks [41–46], but gauge equivariance did not play a role in these papers. Gauge equivariance of neural networks was first discussed in [47, 48]. In short, gauge symmetry can be built into the model by requiring that the map implemented by the neural network commutes with local gauge transformations. As a result, the neural network does not need to learn this symmetry and can achieve the same expressivity with fewer weights. Gauge-equivariant neural networks were constructed to generate gauge-field ensembles in several lattice field theories in [49–51]. Neural networks that do not explicitly preserve gauge equivariance were used as preconditioners in a two-dimensional U(1) lattice gauge theory in [52]. Reference [53] demonstrated that gauge-equivariant neural networks can approximate any gauge-equivariant function on the lattice. In Ref. [54] the equivariance of neural networks was extended to global lattice symmetries, and group-equivariant pooling layers were discussed.

This paper is structured as follows. In Sec. II we describe our coarsening approach using gauge-equivariant pooling and unpooling layers. In Sec. III we provide de-

tails of the Wilson-clover Dirac spectrum on a gauge configuration with nonzero topological charge. In Sec. IV we discuss the training strategy for the (un)pooling layers, and in Sec. V we show that the models resolve critical slowing down. In Sec. VI we summarize our results and provide an outlook on our future research program.

II. GAUGE-EQUIVARIANT COARSENING

In the following, we build on notation defined in Ref. [12] but introduce an explicitly gauge-equivariant coarsening procedure using gauge-equivariant pooling and unpooling layers that are combined with subsampling layers.

A. Review of notation and coarse-grid vector space

We consider a d-dimensional space-time lattice, the fine grid, and denote the set of its sites by S. We define a field $\varphi : S \to V_I$, $x \mapsto \varphi(x)$ on the fine grid with internal vector space

$$V_I = V_G \otimes V_{\bar{G}} \,, \tag{1}$$

where V_G is a gauge vector space and $V_{\bar{G}}$ is a non-gauge vector space, respectively. The set of such fields is denoted by $\mathcal{F}_\varphi$. Under a gauge transformation $\Omega : S \to \mathrm{End}(V_G)$, $x \mapsto \Omega(x)$, the fields transform as $\varphi(x) \to \Omega(x)\varphi(x)$. Furthermore, we consider gauge fields $U_\mu : S \to \mathrm{End}(V_G)$, $x \mapsto U_\mu(x)$, where $\mu \in \{1, \ldots, d\}$. In the case of QCD, $U_\mu(x) \in \mathrm{SU}(3) \subset \mathrm{End}(V_G)$. We will use U as a short-hand notation for the tuple $(U_1, \ldots, U_d)$.

We also consider a d-dimensional coarse grid with set of sites $\tilde{S}$. We define fields on the coarse grid $\tilde{\varphi} : \tilde{S} \to \tilde{V}_I$, $y \mapsto \tilde{\varphi}(y)$ with internal vector space $\tilde{V}_I$. The set of such coarse fields is denoted by $\mathcal{F}_{\tilde{\varphi}}$. In contrast to Ref. [12]

$$\tilde{V}_I = V_G \otimes \tilde{V}_{\bar{G}} \tag{2}$$

i.e., in the current work the local gauge space on the coarse grid is the same as on the fine grid.

As in Ref. [12], we define a block map $B : \tilde{S} \to \mathcal{P}(S)$, where $\mathcal{P}$ denotes the power set. We also define a map

$$B_r : \tilde{S} \to S, y \mapsto B_r(y) \tag{3}$$

that selects for each site y on the coarse grid a reference site $B_r(y)$ on the fine grid. In the following, we only consider maps B_r for which $B_r(y) \in B(y)$. The coarse fields shall transform as

$$\tilde{\varphi}(y) \to \tilde{\Omega}(y)\tilde{\varphi}(y) \tag{4}$$

with

$$\tilde{\Omega}(y) = \Omega(B_r(y)) \tag{5}$$

under gauge transformations Ω. For a related discussion of gauge-equivariant blocking schemes, see, e.g., Ref. [25].
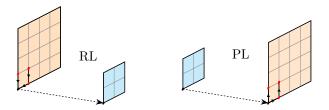

FIG. 1. Graphical representation of restriction layer (left) and prolongation layer (right) for a single feature. The input and output features are represented by the planes, and the layers are represented by the paths drawn and the arrow mapping the input to the output feature. The reference site is drawn in black.

B. Restriction and prolongation layers

The restriction layer (RL) can be written as the composition of a pooling layer (Pool) and a subsampling layer (SubSample),

$$\mathrm{RL} = \mathrm{SubSample} \circ \mathrm{Pool} \,. \tag{6}$$

The pooling layer $\mathrm{Pool} : \mathcal{F}_\varphi \to \mathcal{F}_\varphi$, $\varphi \mapsto \mathrm{Pool}\varphi$ is given by

$$\mathrm{Pool}\varphi(x) = \sum_{q \in Q} W_q(x) T_q \varphi(x) \,. \tag{7}$$

In the following we describe the elements of this equation in detail. The sum is over couples (i.e., two-tuples) $q = (p, \bar{U})$ that consist of a path p and a gauge field $\bar{U}$. A path p is defined as a sequence of hops without reference to a starting or end point. A set of paths P shall be called "complete" if it connects every site in $B(y)$ to $B_r(y)$ exactly once. A complete set of paths therefore always has $|B(y)|$ elements, where $|X|$ denotes the cardinality of a set X. In the current work, we only consider couples with $|Q| = n|B(y)|$ and $n \in \mathbb{N}^+$ such that n prescriptions to construct the gauge field are combined with n prescriptions to construct a complete set of paths.

The pooling layer is parametrized by weights $W_q(x) \in \mathrm{End}(V_{\bar{G}})$. In the context of the current paper the $W_q(x)$ are spin matrices.

Finally, the operator T_q for $q = (p, \bar{U})$ is the parallel-transport operator $T_p : \mathcal{F}_\varphi \to \mathcal{F}_\varphi$, $\varphi \mapsto T_p\varphi$ defined in Ref. [12] with gauge fields U replaced by $\bar{U}$.[1] The gauge fields $\bar{U}$ entering T_p do not have to be the original fine-grid gauge links U as long as they transform in the usual way, i.e., as

$$\bar{U}_\mu(x) \to \Omega(x)\bar{U}_\mu(x)\Omega^\dagger(x + \hat{\mu}) \tag{8}$$

under gauge transformations Ω. We will make use of this freedom in this work.

---

[1] In Eq. (7), $T_q\varphi(x)$ means that the field $T_q\varphi$ is evaluated at x.
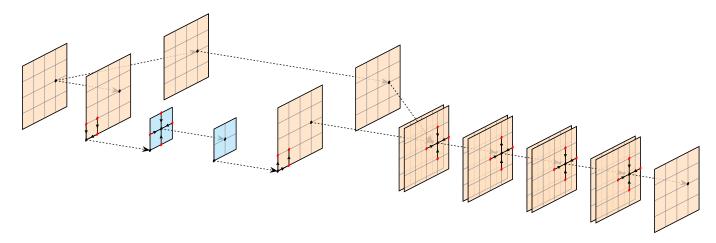
FIG. 2. The two-level multigrid model studied in this work. The model is similar to the one studied in Ref. [12], but explicitly gauge-equivariant pooling and unpooling layers are used in the current work for the restriction and prolongation layers. The coarse-grid layer is limited by the blue features. This layer and the last four layers are LPTC layers introduced in Ref. [12].

The subsampling layer SubSample: $\mathcal{F}_\varphi \to \mathcal{F}_{\tilde\varphi}$, $\varphi \mapsto$ SubSampleφ is defined by

$$\text{SubSample}\varphi(y) = \varphi(B_r(y)) \qquad (9)$$

for a given choice of reference-point map B_r defined in Eq. (3). This construction therefore satisfies Eq. (4) with $\tilde\varphi = \text{RL}\varphi$ for a given $\varphi \in \mathcal{F}_\varphi$. For a discussion of a general group-equivariant pooling layer, see Ref. [54].

The prolongation layer (PL) is simply defined as

$$\text{PL} = \text{Pool}^\dagger \circ \text{SubSample}^\dagger, \qquad (10)$$

where the dagger of an operator O is defined in the usual way by requiring $\varphi_1^\dagger O \varphi_2 = (\varphi_2^\dagger O^\dagger \varphi_1)^*$ for arbitrary φ_1 and φ_2. Note that the couples and weights of a restriction and prolongation layer can in principle be chosen independently. The models studied in this work, however, use the same couples and weights for both RL and PL so that $\text{PL} = \text{RL}^\dagger$.[2]

A graphical representation of the restriction and prolongation layers is given in Fig. 1. The pooling layer is a generalization of the local parallel-transport convolution (LPTC) layer introduced in Ref. [12]. However, one would typically implement the combined RL directly to avoid unnecessary computation of feature elements that will be discarded by the subsequent subsampling layer. This can be done efficiently by precomputing, for each complete set of paths, a field $S \to \text{End}(V_G)$ that is used in combination with a reduction operation within each block. We provide such implementations of both RL and PL in the Grid Python Toolkit (GPT) [56].

We note that the construction of similar restriction and prolongation operations has a long history, see, e.g., [20, 25, 35].

---

[2] In the context of a multigrid solver, Ref. [55] calls this the variational choice because it follows from a variational principle.

C. Coarsening of the gauge fields

In the current work, we preserve the general model structure introduced in Ref. [12]. However, we replace the restriction and prolongation layers with ones based on gauge-equivariant pooling and unpooling layers, see Fig. 2. This replacement introduces an explicit gauge degree of freedom on the coarse grid so that the coarse-grid layer can be constructed in an explicitly gauge-equivariant manner. For this layer we need coarse gauge fields $\tilde U$.

The gauge transformation property of coarse fields given in Eq. (4) is consistent with gauge fields on the coarse grid that perform a parallel transport between reference sites $B_r(y)$ and $B_r(y')$ on the fine grid, where y and y' are neighboring sites on the coarse grid. Such gauge fields must transform as

$$\tilde U_\mu(y) \to \tilde\Omega(y)\tilde U_\mu(y)\tilde\Omega^\dagger(y + \hat\mu) \qquad (11)$$

under gauge transformations. We investigate two choices for the $\tilde U_\mu$ in this work.

The first choice is to connect $B_r(y)$ and $B_r(y')$ using the shortest path on the fine grid connecting both points. In this work, we use a block map B such that $B(y)$ is given by a Cartesian product of neighboring sites in each dimension, and a fixed reference site B_r within each block so that the shortest path is unique and aligns with a coordinate axis. We then always have

$$B_r(y') - B_r(y) = b\hat\mu \qquad (12)$$

with unit vector $\hat\mu$ in direction μ and $b \in \mathbb{N}^+$. The coarse-grid gauge field $\tilde U_\mu(y)$ corresponding to this pair of reference points is then simply

$$\tilde U_\mu(y) = U_\mu(B_r(y)) \cdots U_\mu(B_r(y) + (b-1)\hat\mu) \qquad (13)$$

with fine-grid gauge links U_μ. We will refer to this choice as the "plain coarse-link model."

The second choice is based on the Galerkin coarse-grid operator

$$\tilde{D} = \mathrm{RL} \circ D \circ \mathrm{PL} \tag{14}$$

with gauge-equivariant fine-grid operator D. For the purpose of the current paper, D is the Wilson-clover Dirac operator (for the precise definition see Ref. [12]). We then simply define

$$\tilde{U}_\mu(y) = \tilde{D}(y, y + \hat{\mu}), \tag{15}$$

which transforms as in Eq. (11) since $\tilde{D}(y, y')$ transforms to $\tilde{\Omega}(y)\tilde{D}(y, y')\tilde{\Omega}^\dagger(y')$ under gauge transformations Ω. We refer to this choice as the "Galerkin model." Note that in the Galerkin model the coarse gauge links will depend on the weights in the RL and PL. In the Galerkin model $\tilde{U}_\mu(y) \in \mathrm{End}(\tilde{V}_I)$, while $\tilde{U}_\mu(y) \in \mathrm{End}(V_G)$ in the plain coarse-link model. Both are acceptable in the context of the gauge-equivariant coarse-grid LPTC layer in Fig. 2 as long as Eq. (11) is satisfied.

We again note that there is a rich history of related work, see, e.g., Refs. [21, 38, 57–60]. As in these works, our coarse gauge fields defined by Eq. (15) are, in general, no longer elements of the original gauge group. While this is not a problem of principle, Refs. [21, 27] found better performance of the multigrid algorithm if the coarse gauge fields are projected back to the original gauge group. We plan to investigate this possibility in future work. We also note that there is an alternative way to define the coarse gauge fields using the pooling and subsampling layers introduced in Sec. II B and applying them to the gauge links between the blocks, see, e.g., Ref. [38]. We did not implement this alternative because it does not increase the expressivity of the model compared to Eq. (15).

III. DIRAC SPECTRUM AND TOPOLOGY

As in Ref. [12], we have generated quenched Wilson gauge configurations with $8^3 \times 16$ lattice sites for $\beta = 6$ and attempt to precondition the Dirac equation for the Wilson-clover Dirac operator with $c_{\mathrm{sw}} = 1$. In order to provide an even more challenging setup for the preconditioner models, we select gauge configurations with topological charge $Q = 1$ defined via the five-loop enhanced definition of Ref. [61] after cooling the gauge fields by applying the Wilson flow [62] with flow time $t = 10$.[3] The Dirac operator has an eigenvalue with vanishing imaginary part and real part very close to the lower edge of the spectrum, see Fig. 3. In this case, we expect critical slowing down to be clearly visible as the quark mass is tuned to criticality.
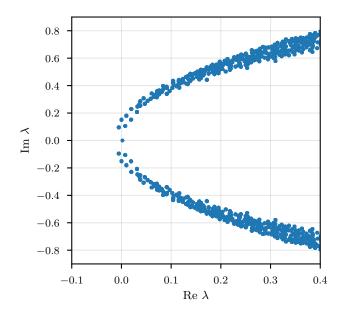
---

[3] The measured value for the configuration used in this work is $Q = 0.998$.

FIG. 3. Smallest eigenvalues λ of the Wilson-clover Dirac operator with mass $m = -0.5645$ and $c_{\mathrm{sw}} = 1$ on a pure-Wilson-gauge configuration with topological charge $Q = 1$, $\beta = 6$, and $8^3 \times 16$ lattice sites. The mass m is tuned to near criticality for the experiments in this work.

IV. TRAINING STRATEGY

In the following we describe our training strategy for the preconditioner model shown in Fig. 2. We perform the training in two steps.

In the first step, we train only the restriction and prolongation layers. One may naturally consider to train PL ∘ RL as an autoencoder with training vectors sampled from the low-mode space of D. We find that this strategy by itself is not sufficient to obtain an efficient model. Instead, we also train PL ∘ RL to act as a projector onto the low-mode space, i.e., it should project high modes to zero. Furthermore, we found that it is beneficial to approximately preserve the property RL ∘ PL = $\mathbb{1}$. We also found that restricting PL = RL† by using the same couples $q = (p, \bar{U})$ and the same weights $W_q(x)$ for the restriction and prolongation layers did not reduce the performance of the model, and therefore we adopt this choice for simplicity.

We implement this strategy by using the cost function

$$\begin{aligned} C = |\mathrm{PL} \circ \mathrm{RL}v_\ell - v_\ell|^2 &+ |\mathrm{PL} \circ \mathrm{RL}v_h - P_\ell v_h|^2 \\ &+ |\mathrm{RL} \circ \mathrm{PL}v_c - v_c|^2 \end{aligned} \tag{16}$$

with two fine-grid vectors v_ℓ and v_h and one coarse-grid vector v_c. For each training step new random vectors v_ℓ, v_h, v_c are chosen according to the following procedure. For v_ℓ we select a random element of $\{u_1, \ldots, u_s\}$ of the near-null space vectors u_i defined in Ref. [12] with $s \in \mathbb{N}^+$. For v_h and v_c we take random vectors with elements normally distributed about zero. The low-mode

projector

$$P_\ell = W^\dagger W \quad (17)$$

with W defined in Eq. (31) of [12] is using the same set of near-null vectors $\{u_1, \ldots, u_s\}$. All vectors v_ℓ, v_h, and v_c are normalized to unit length before being used in the cost function. Note that $P_\ell v_\ell = v_\ell$ by construction so that we can also write the cost function in the more symmetric way

$$C = |\text{PL} \circ \text{RL}v_\ell - P_\ell v_\ell|^2 + |\text{PL} \circ \text{RL}v_h - P_\ell v_h|^2$$
$$+ |\text{RL} \circ \text{PL}v_c - v_c|^2 . \quad (18)$$

This training procedure provides the gauge-invariant spin matrices $W_q(x)$ for a given gauge configuration. While the current training strategy does not reduce the overall cost compared to the multigrid model studied in Ref. [12], we will study constructing the gauge-invariant $W_q(x)$ directly from a given gauge field U using gauge-invariant models [53] in future work. The local features of $W_q(x)$ may be related to features of the local energy density, topological charge density, and general Wilson loops so that no retraining may be needed for a different gauge configuration of the same ensemble.

In the second step, we use the trained RL and PL in the model M of Fig. 2 and train the model with frozen pooling-layer weights using the same cost function as in Ref. [12],

$$C = |Mb_h - u_h|^2 + |Mb_\ell - u_\ell|^2 \quad (19)$$

with $b_h = Dv_1$, $u_h = v_1$, $b_\ell = v_2$, and $u_\ell = D^{-1}v_2$. Here, v_1 and v_2 are random vectors normalized such that $|b_h| = |b_\ell| = 1$. After this procedure, we can also continue to train the model without freezing the pooling-layer weights. However, no benefit was observed from this refinement.

V. MODEL DETAILS AND RESULTS

In this section we demonstrate the performance of the models we studied with focus on removing the critical slowing down in solving the Dirac equation when the mass parameter m is tuned towards criticality.

For concreteness, we use a coarse grid of size $2^3 \times 4$ such that $|B(y)| = 4^4$, and $s = 4$. Note that in Ref. [12] we used $s = 12$. However, for the case at hand $s = 4$ was sufficient to obtain a well-performing model.

For the pooling layers, we found that using gauge fields which are smeared differently depending on the set of paths works well. Concretely, we use 9 different gauge fields $\bar{U}^{(i)}$ with $i = 1, \ldots, 9$. We construct the $\bar{U}^{(i)}$ by applying $i(i-1)/2$ steps of $\rho = 0.1$ stout smearing [63] to the unsmeared gauge fields U. For fixed i, we define paths $p^{(ij)}$ that connect all elements of $B(y)$, enumerated by $j = 1, \ldots, |B(y)|$, to the reference site $B_r(y)$. For different i we use different prescriptions for the paths $p^{(ij)}$,
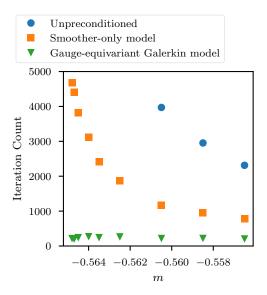

FIG. 4. Outer iteration count of unpreconditioned and preconditioned solvers as a function of the quark mass. The gauge-equivariant Galerkin model completely removes the critical slowing down as the mass is tuned to criticality.

and then use the couples $q_{ij} = (p^{(ij)}, \bar{U}^{(i)})$ in Eq. (7). We define four different prescriptions $\hat{p}_1, \ldots, \hat{p}_4$ in the following and set $p^{(ij)} = \hat{p}_{i \bmod 4}^{(j)}$.

For all prescriptions we select the reference site to be the origin of each block. For the first block the reference site corresponds to coordinate $(1, 1, 1, 1)$. The starting site for path p is denoted by $(x_1+1, x_2+1, x_3+1, x_4+1)$. Then the first prescription to construct the paths is to use $T_p = H_{-4}^{x_4} H_{-3}^{x_3} H_{-2}^{x_2} H_{-1}^{x_1}$ with hopping operator $H_{\pm\mu}$ defined in Ref. [12]. The second prescription is $T_p = H_{-1}^{x_1} H_{-2}^{x_2} H_{-3}^{x_3} H_{-4}^{x_4}$. The third and fourth prescriptions modify the first and second prescription, respectively, by permuting the hops in a way that, to the degree possible, at most one hop in one direction is performed at a time. A concrete example for $x_1 = 2$, $x_2 = 2$, $x_3 = 1$, $x_4 = 1$ is $T_p = H_{-2}H_{-1}H_{-4}H_{-3}H_{-2}H_{-1}$ for the third prescription and $T_p = H_{-1}H_{-2}H_{-1}H_{-2}H_{-3}H_{-4}$ for the fourth prescription.

We investigated many additional choices for reference sites, paths, and gauge fields to be used in the pooling-layer construction. However, the setup just described proved to perform well while still being relatively simple.

We then solve the Dirac equation with and without preconditioning and study the iteration count of the outer FGMRES [64] solver to 10^{-8} precision as a function of the quark mass m. In Fig. 4 we compare the outer iteration count of the unpreconditioned solver with the smoother-only model of Ref. [12] and the new gauge-equivariant Galerkin model. We find that in the smoother-only model critical slowing down is still visible, while it is completely absent in the gauge-equivariant Galerkin model. In Fig. 5 we compare the original multigrid model of
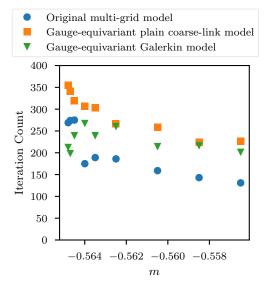
FIG. 5. Comparison of multigrid models studied in this work and the original multigrid model of Ref. [12]. The gauge-equivariant Galerkin model performs very well even for masses near criticality. The plain coarse-link model shows a mild increase in outer iteration count near criticality.

Ref. [12] with the gauge-equivariant Galerkin model and with the gauge-equivariant plain coarse-link model. We find that the original model and the gauge-equivariant Galerkin model perform best, while the plain coarse-link model indicates a small remaining signature of critical slowing down. Note that there is some randomness in the training procedure that explains the performance fluctuations between neighboring mass points for a given model.

VI. SUMMARY AND OUTLOOK

The current work is part of a larger research program based on gauge-equivariant multigrid neural networks.

In our first paper [12] we demonstrated that a state-of-the-art multigrid preconditioner can be learned efficiently by gauge-equivariant neural networks. The restriction and prolongation layers of Ref. [12] were, however, manually constructed by traditional methods to find near-null-space vectors.

In the current work, we replaced this construction by gauge-equivariant pooling and unpooling layers that are learned for a given gauge configuration. We demonstrated that such models can eliminate critical slowing down and perform as well as traditional multigrid models. The pooling and unpooling layers are parametrized by gauge-invariant spin matrices, which in turn can be learned by models such as discussed in Ref. [53]. The construction of such models, including a detailed study of transfer learning, is left for future work. If successful, such models promise to drastically reduce the setup cost in multigrid preconditioners and may therefore play an important role in improving the performance of gauge-generation algorithms such as HMC [65] or flow-based models [49–51, 66, 67].

Another important topic left for future studies is the construction of multigrid models for operators with a more challenging spectrum, such as domain-wall fermions.

Finally, gauge-equivariant multigrid models should also be able to learn to directly approximate complex hadronic correlation functions without constructing them from intermediate approximations of propagators. Such direct approximations can then be used to reduce statistical noise without introducing bias [68]. We will explore this application of gauge-equivariant multigrid models in future work as well.

* Corresponding author; christoph.lehner@ur.de

[1] M. Lüscher, Solution of the Dirac equation in lattice QCD using a domain decomposition method, Comput. Phys. Commun. **156**, 209 (2004), arXiv:hep-lat/0310048.

[2] M. Lüscher, Local coherence and deflation of the low quark modes in lattice QCD, JHEP **07**, 081, arXiv:0706.2298 [hep-lat].

[3] 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), arXiv:0707.4018 [hep-lat].

[4] 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), arXiv:1005.3043 [hep-lat].

[5] 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 (2014), arXiv:1303.1377 [hep-lat].

[6] P. A. Boyle, Hierarchically deflated conjugate gradient, (2014), arXiv:1402.2585 [hep-lat].

[7] J. Brannick, A. Frommer, K. Kahl, B. Leder, M. Rottmann, and A. Strebel, Multigrid Preconditioning for the Overlap Operator in Lattice QCD, Numer. Math. **132**, 463 (2016), arXiv:1410.7170 [hep-lat].

[8] A. Yamaguchi and P. Boyle, Hierarchically deflated conjugate residual, PoS **LATTICE2016**, 374 (2016), arXiv:1611.06944 [hep-lat].

[9] R. C. Brower, M. A. Clark, A. Strelchenko, and E. Weinberg, Multigrid algorithm for staggered lattice fermions, Phys. Rev. D **97**, 114513 (2018), arXiv:1801.07823 [hep-lat].

[10] R. C. Brower, M. A. Clark, D. Howarth, E. S. Weinberg, Multigrid for chiral lattice fermions: Domain wall, Phys. Rev. D **102**, 094517 (2020), arXiv:2004.07732 [hep-lat].

[11] P. Boyle and A. Yamaguchi, Comparison of Domain Wall Fermion Multigrid Methods (2021), arXiv:2103.05034 [hep-lat].

[12] C. Lehner and T. Wettig, Gauge-equivariant neural networks as preconditioners in lattice QCD (2023), arXiv:2302.05419 [hep-lat].

[13] J. Goodman and A. D. Sokal, Multigrid Monte Carlo Method for Lattice Field Theories, Phys. Rev. Lett. **56**, 1015 (1986).

[14] A. Hulsebos, J. Smit, and J. C. Vink, Multigrid Monte Carlo in an external U(1) gauge field, Nucl. Phys. B Proc. Suppl. **9**, 512 (1989).

[15] A. Hulsebos, J. Smit, and J. C. Vink, Multigrid Monte Carlo for a Bose Field in an External Gauge Field, Nucl. Phys. B **331**, 531 (1990).

[16] A. Hulsebos, J. Smit, and J. C. Vink, Multigrid inversion of the staggered fermion matrix, Nucl. Phys. B Proc. Suppl. **20**, 94 (1991).

[17] A. Hulsebos, J. Smit, and J. C. Vink, Multigrid inversion of the staggered fermion matrix with U(1) and SU(2) gauge fields, Int. J. Mod. Phys. C **3**, 161 (1991).

[18] J. C. Vink, Multigrid inversion of staggered and Wilson fermion operators with SU(2) gauge fields in two dimensions, Phys. Lett. B **272**, 81 (1991).

[19] A. Hulsebos, J. Smit, and J. C. Vink, Multigrid inversion of lattice fermion operators, Nucl. Phys. B **368**, 379 (1992).

[20] R. Ben-Av, A. Brandt, and S. Solomon, The Fermionic Matrix, Instantons, Zero Modes and Multigrid, Nucl. Phys. B **329**, 193 (1990).

[21] R. Ben-Av, A. Brandt, M. Harmatz, E. Katznelson, P. G. Lauwers, S. Solomon, and K. Wolowesky, Fermion simulations using parallel transported multigrid, Phys. Lett. B **253**, 185 (1991).

[22] M. Harmatz, P. G. Lauwers, R. Ben-Av, A. Brandt, E. Katznelson, S. Solomon, and K. Wolowesky, Parallel transported multigrid and its application to the Schwinger model, Nucl. Phys. B Proc. Suppl. **20**, 102 (1991).

[23] P. G. Lauwers, R. Ben-Av, and S. Solomon, Inverting the Dirac matrix for SU(2) lattice gauge theory by means of parallel transported multigrid, Nucl. Phys. B **374**, 249 (1992).

[24] S. Solomon and P. G. Lauwers, Parallel transported multigrid beats conjugate gradient, Int. J. Mod. Phys. C **3**, 149 (1992).

[25] R. Ben-Av, M. Harmatz, P. G. Lauwers, and S. Solomon, Parallel transported multigrid for inverting the Dirac operator: Variants of the method and their efficiency, Nucl. Phys. B **405**, 623 (1993).

[26] M. Harmatz, S. Solomon, P. Lauwers, and T. Wittlich, Visual study of zero modes role in PTMG convergence, Nucl. Phys. B Proc. Suppl. **30**, 192 (1993).

[27] P. G. Lauwers and T. Wittlich, Inversion of the fermion matrix in lattice QCD by means of parallel transported multigrid (PTMG), Int. J. Mod. Phys. C **4**, 609 (1993).

[28] P. G. Lauwers and T. Wittlich, Parallel transported multigrid (PTMG) for inverting the Dirac operator in SU(3) lattice gauge theory, Nucl. Phys. B Proc. Suppl. **30**, 261 (1993).

[29] R. C. Brower, K. J. M. Moriarty, C. Rebbi, and E. Vicari, Multigrid propagators in the presence of disordered U(1) gauge fields, Phys. Rev. D **43**, 1974 (1991).

[30] R. C. Brower, C. Rebbi, and E. Vicari, Projective multigrid method for propagators in lattice gauge theory, Phys. Rev. D **43**, 1965 (1991).

[31] R. C. Brower, C. Rebbi, and E. Vicari, Non-Abelian Projective Multigrid for Lattice Gauge Theory, Phys. Rev. Lett. **66**, 1263 (1991).

[32] R. C. Brower, R. G. Edwards, C. Rebbi, and E. Vicari, Projective multigrid for Wilson fermions, Nucl. Phys. B **366**, 689 (1991).

[33] T. Kalkreuter, Projective block spin transformations in lattice gauge theories, Nucl. Phys. B **376**, 637 (1992).

[34] T. Kalkreuter, Ground state projection multigrid for propagators in four-dimensional SU(2) gauge fields, Phys. Lett. B **276**, 485 (1992).

[35] T. Kalkreuter, G. Mack, and M. Speh, Blockspin and multigrid for staggered fermions in nonAbelian gauge fields, Int. J. Mod. Phys. C **3**, 121 (1992).

[36] T. Kalkreuter, Improving multigrid and conventional relaxation algorithms for propagators, Int. J. Mod. Phys. C **3**, 1323 (1992), arXiv:hep-lat/9207016.

[37] T. Kalkreuter, Multigrid for propagators of staggered fermions in four-dimensional SU(2) gauge fields, Nucl. Phys. B Proc. Suppl. **30**, 257 (1993), arXiv:hep-lat/9211008.

[38] T. Kalkreuter, Multigrid methods for the computation of propagators in gauge fields, Int. J. Mod. Phys. C **5**, 629 (1994), arXiv:hep-lat/9212021.

[39] T. Kalkreuter, Idealized multigrid algorithm for staggered fermions, Phys. Rev. D **48**, R1926 (1993), arXiv:hep-lat/9304004.

[40] M. Bäker, T. Kalkreuter, G. Mack, and M. Speh, Neural multigrid for gauge theories and other disordered systems, Int. J. Mod. Phys. C **4**, 239 (1993), arXiv:hep-lat/9209021.

[41] A. Katrutsa, T. Daulbaev, and I. Oseledets, Deep Multigrid: learning prolongation and restriction matrices (2017), arXiv:1711.03825 [math.NA].

[42] D. Greenfeld, M. Galun, R. Basri, I. Yavneh, and R. Kimmel, Learning to Optimize Multigrid PDE Solvers, in *Proceedings of the 36th International Conference on Machine Learning* (2019) pp. 2415–2423, arXiv:1902.10248 [math.NA].

[43] I. Luz, M. Galun, H. Maron, R. Basri, and I. Yavneh, Learning algebraic multigrid using graph neural networks, in *Proceedings of the 37th International Conference on Machine Learning* (2020) pp. 6489–6499, arXiv:2003.05744 [cs.LG].

[44] M. Eliasof, J. Ephrath, L. Ruthotto, and E. Treister, MGIC: Multigrid-in-Channels Neural Network Architectures (2020), arXiv:2011.09128 [cs.CV].

[45] R. Huang, R. Li, and Y. Xi, Learning optimal multigrid smoothers via neural networks, SIAM J. Sci. Comput. , S199 (2021), arXiv:2102.12071 [math.NA].

[46] A. van Betteray, M. Rottmann, and K. Kahl, MGiaD: Multigrid in all dimensions. Efficiency and robustness by coarsening in resolution and channel dimensions (2022), arXiv:2211.05525 [cs.CV].

[47] T. Cohen and M. Welling, Group Equivariant Convolu-

tional Networks, in *Proceedings of The 33rd International Conference on Machine Learning* (2016) pp. 2990–2999, arXiv:1602.07576 [cs.LG].

[48] T. S. Cohen, M. Weiler, B. Kicanaoglu, and M. Welling, Gauge Equivariant Convolutional Networks and the Icosahedral CNN, in *Proceedings of the 36th International Conference on Machine Learning* (2019) pp. 1321–1330, arXiv:1902.04615 [cs.LG].

[49] G. Kanwar, M. S. Albergo, D. Boyda, K. Cranmer, D. C. Hackett, S. Racanière, D. J. Rezende, and P. E. Shanahan, Equivariant flow-based sampling for lattice gauge theory, Phys. Rev. Lett. **125**, 121601 (2020), arXiv:2003.06413 [hep-lat].

[50] D. Boyda, G. Kanwar, S. Racanière, D. J. Rezende, M. S. Albergo, K. Cranmer, D. C. Hackett, and P. E. Shanahan, Sampling using $SU(N)$ gauge equivariant flows, Phys. Rev. D **103**, 074504 (2021), arXiv:2008.05456 [hep-lat].

[51] R. Abbott *et al.*, Gauge-equivariant flow models for sampling in lattice field theories with pseudofermions, Phys. Rev. D **106**, 074506 (2022), arXiv:2207.08945 [hep-lat].

[52] S. Calì, D. C. Hackett, Y. Lin, P. E. Shanahan, and B. Xiao, Neural-network preconditioners for solving the Dirac equation in lattice gauge theory, (2022), arXiv:2208.02728 [hep-lat].

[53] M. Favoni, A. Ipp, D. I. Müller, and D. Schuh, Lattice Gauge Equivariant Convolutional Neural Networks, Phys. Rev. Lett. **128**, 032003 (2022), arXiv:2012.12901 [hep-lat].

[54] J. Aronsson, D. I. Müller, and D. Schuh, Geometrical aspects of lattice gauge equivariant convolutional neural networks (2023), arXiv:2303.11448 [hep-lat].

[55] J. Goodman and A. D. Sokal, Multigrid Monte Carlo method: Conceptual foundations, Phys. Rev. D **40**, 2035 (1989).

[56] C. Lehner et al., Grid Python Toolkit (GPT).

[57] H. B. Nielsen and A. Patkos, Effective Dielectric Theory from QCD, Nucl. Phys. B **195**, 137 (1982).

[58] T. Balaban, Regularity and Decay of Lattice Green's Functions, Commun. Math. Phys. **89**, 571 (1983).

[59] G. Mack, Dielectric lattice gauge theory, Nucl. Phys. B **235**, 197 (1984).

[60] A. D. Sokal, Some comments on multigrid methods for computing propagators, Phys. Lett. B **317**, 399 (1993), arXiv:hep-lat/9307020.

[61] P. de Forcrand, M. Garcia Perez, and I.-O. Stamatescu, Topology of the SU(2) vacuum: A Lattice study using improved cooling, Nucl. Phys. B **499**, 409 (1997), arXiv:hep-lat/9701012.

[62] M. Lüscher, Properties and uses of the Wilson flow in lattice QCD, JHEP **08**, 071, [Erratum: JHEP 03, 092 (2014)], arXiv:1006.4518 [hep-lat].

[63] C. Morningstar and M. J. Peardon, Analytic smearing of SU(3) link variables in lattice QCD, Phys. Rev. D **69**, 054501 (2004), arXiv:hep-lat/0311018.

[64] Y. Saad, A Flexible Inner-Outer Preconditioned GMRES Algorithm, SIAM J. Sci. Comput. **14**, 461 (1993).

[65] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, Hybrid Monte Carlo, Phys. Lett. B **195**, 216 (1987).

[66] R. Abbott *et al.*, Aspects of scaling and scalability for flow-based sampling of lattice QCD, (2022), arXiv:2211.07541 [hep-lat].

[67] R. Abbott *et al.*, Sampling QCD field configurations with gauge-equivariant flow models, PoS **LATTICE2022**, 036 (2023), arXiv:2208.03832 [hep-lat].

[68] E. Shintani, R. Arthur, T. Blum, T. Izubuchi, C. Jung, and C. Lehner, Covariant approximation averaging, Phys. Rev. D **91**, 114511 (2015), arXiv:1402.0244 [hep-lat].