# Acceleration of the Arnoldi method and real eigenvalues of the non-Hermitian Wilson-Dirac operator

Georg Bergner[1,*] and Jaïr Wuilloud[2,†]

[1] *Institut für Theoretische Physik, Westfälische Wilhelms-Universität Münster, Wilhelm-Klemm-Str. 9, D-48149 Münster, Germany*

[2] *Albert Einstein Institute for Fundamental Physics Institute for Theoretical Physics University of Bern, Sidlerstr. 5, CH-3012 Bern, Switzerland*

(Dated: August 20, 2021)

## Abstract

In this paper, we present a method for the computation of the low-lying real eigenvalues of the Wilson-Dirac operator based on the Arnoldi algorithm. These eigenvalues contain information about several observables. We used them to calculate the sign of the fermion determinant in one-flavor QCD and the sign of the Pfaffian in $\mathcal{N} = 1$ super Yang-Mills theory. The method is based on polynomial transformations of the Wilson-Dirac operator, leading to considerable improvements of the computation of eigenvalues. We introduce an iterative procedure for the construction of the polynomials and demonstrate the improvement in the efficiency of the computation. In general, the method can be applied to operators with a symmetric and bounded eigenspectrum.



* g.bergner@uni-muenster.de

† wuilloud@gmail.com

## I. INTRODUCTION

The Wilson-Dirac operator $D_W$, which is used in many recent lattice simulations to represent the fermionic part of the discretized action, has the following form

$$(D_W)_{n,\alpha;m,\beta} = \delta_{n,m}\delta_{\alpha,\beta} - \kappa \sum_{\mu=1}^{4} \left[ (1-\gamma_\mu)_{\alpha,\beta} U_\mu(n) \delta_{n+\mu,m} + (1+\gamma_\mu)_{\alpha,\beta} U_\mu^\dagger(n-\mu) \delta_{n-\mu,m} \right] . \quad (1)$$

Here $n, m$ denote points in a four-dimensional hypercubical space-time lattice, $\alpha, \beta$ are Dirac indices, $\mu = 1, 2, 3, 4$ labels the positive directions and $\gamma_\mu$ are the Dirac matrices. The hopping parameter $\kappa$ is related to the bare fermion mass; in particular $\kappa$ increases for decreasing fermion masses. The link variables $U_\mu(n)$ are associated with the links connecting neighboring lattice points and represent the gauge field. In our investigations [1, 2] the gauge field was in the fundamental representation of SU(3) for QCD with one quark flavor (one-flavor QCD) and in the adjoint representation of SU(2) (real $3 \times 3$ matrices) for supersymmetric Yang-Mills theory. The method presented here is, however, not restricted to a specific gauge group and can be applied also to other fermion operators.

In the free theory, the eigenspectrum of $D_W$ can be decomposed into a physical branch, consisting of the smallest eigenvalues, and the doublers, which become irrelevant in the continuum limit [3]. Such a clear distinction of relevant and irrelevant parts is not possible in the interacting case. However, the lowest part of the spectrum still contains the most important information. The low eigenvalue part plays a crucial role in spectral decompositions of the fermionic observables [4], and the lowest eigenmodes allow for an acceleration of the inversion by deflation [5].

For several investigations, the Hermitian operator $\gamma_5 D_W$ can be used instead of $D_W$. The corresponding eigenvalue problem can also be solved with other iterative methods, but for the non-normal operator $D_W$ the (restarted) Arnoldi algorithm [6] seems to be the optimal choice.[1]

The importance of the lowest eigenmodes of $D_W$ has been the subject of several recent investigations, e. g. in [8, 9]. Furthermore, their implications on the topology of gauge fields has been studied (e. g. in [10, 11]), even though $D_W$ does not allow for a realization of chiral symmetry on the lattice.

In several cases, numerical simulations of field theories with dynamical fermions require a reweighting of the observables with the sign of the determinant of $D_W$ or of its Pfaffian. This sign can be obtained from the number of negative real eigenmodes [1, 12]. The computation of the reweighting for one-flavor QCD and $\mathcal{N} = 1$ super Yang-Mills theory was the main purpose of our investigations of the spectrum of $D_W$.

On small lattices, the complete set of eigenvalues is accessible (see e. g. Fig. 1). In a more realistic setup, strategies focusing the computation on the relevant small eigenvalues and accelerating the convergence are required. For lattice QCD, a polynomial approach focusing on the low eigenmodes of $D_W$ has been presented in [13]. Within a mathematical framework, other methods based on polynomial transformations have been developed for the computation of a particular sector of a general eigenspectrum [14, 15]. We explain here a new strategy to obtain the lowest real eigenmodes of the Wilson-Dirac operator and show its impact on the efficiency of the computation. Our strategy allowed us to obtain the relevant part of the spectrum on lattices up to a size of $32^3 \times 64$ lattice points.

[1] See, e. g., [7] for a detailed discussion of the effects of the non-normality.

This paper is organized as follows. In the next section we explain the basic idea of the polynomial transformations. In Section III we present our specific strategy to obtain the necessary polynomial. Section IV contains some results and Section V a comparison with other methods. Further mathematical explanations and some practical considerations can be found in Section VI.

## II. ACCELERATION AND FOCUSING OF THE ARNOLDI ALGORITHM

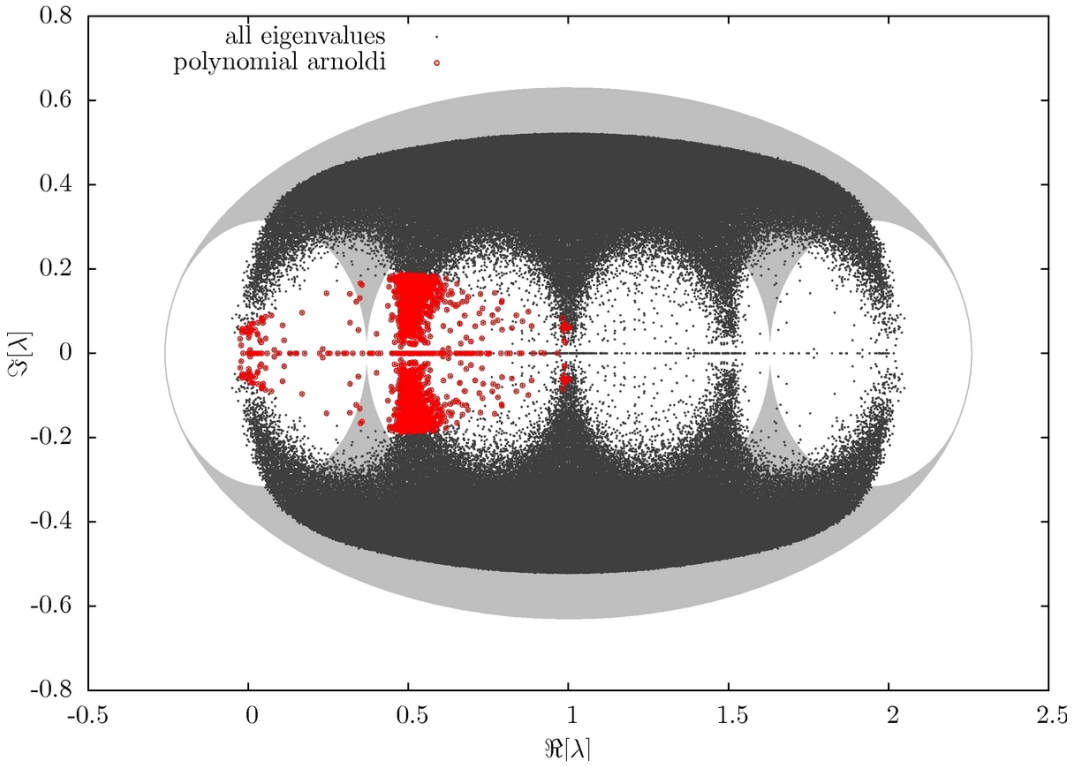



FIG. 1. As an illustration of the method and the form of the spectrum to be expected, this figure shows all eigenvalues of the Wilson-Dirac operator $D_W$ from 29 independent thermalized configurations of a dynamical simulation of the $\mathcal{N} = 1$ super-Yang-Mills theory [2] on a $6^3 \times 8$ lattice ($\beta = 1.6$; $\kappa = 0.1575$). The red points correspond to the eigenvalues computed with the peeling method described below. The gray background shows the region of the eigenvalues in the free theory with the same value of $\kappa$.

Let $\Omega^{(c)}_{D_W}$ be the region that contains all eigenvalues of the operator. We are only interested in a subset of eigenvalues enclosed in a region denoted by $\Omega^{(f)}_{D_W}$. In our case, $\Omega^{(f)}_{D_W}$ is chosen to be a prolate region surrounding all real eigenvalues smaller (or larger) than a certain value (e. g. $\Omega^{(f)}_{D_W} = \{\lambda_i | \, |\Im[\lambda_i]| < \varepsilon, \Re[\lambda_i] \geq x_{\min}\}$ with $\varepsilon$ small).

Fig. 1 shows that the spectrum of $D_W$ contains large regions with a high eigenvalue density and a nonzero imaginary part. For an efficient calculation of the real eigenvalues, it is crucial to exclude these regions and focus the computation on the eigenvalues in $\Omega^{(f)}_{D_W}$. The Arnoldi algorithm computes the eigenvalues starting from those with largest real part.

It calculates the eigenvalues in the region $\Omega^{(a)}_{D_W} = \{\lambda | \Re[\lambda] > x^{(D_W)}_{\rm ar}\}$, where $x^{(D_W)}_{\rm ar}$ depends on the parameters of the algorithm and the eigenvalue distribution of $D_W$.

Hence, a direct computation does not focus efficiently enough on $\Omega^{(f)}_{D_W}$ since a lot of unwanted eigenvalues are calculated. However, an appropriate polynomial transformation $D_W \to p(D_W)$ leads to a better overlap of $p(\Omega^{(f)}_{D_W})$ with $\Omega^{(a)}_{p(D_W)}$. The computation gets focused on the relevant part and a smaller number of unwanted eigenvalues is computed. The eigenvalues of $D_W$ can be obtained from the eigenvalues or eigenvectors of $p(D_W)$.

The second advantage of the polynomial transformation is an acceleration of the Arnoldi computation. The computation of an eigenvalue $\lambda$ converges faster, if this eigenvalue is better separated from the rest of the spectrum (compared to some average distance of the eigenvalues). Therefore, a polynomial minimized on $\Omega^{(c)} \setminus \{\lambda\}$ (with $p(\lambda_i)$ fixed) leads to an acceleration of the computation of $\lambda_i$ (for details cf. [14, 16]). An analytic solution for the absolute minimum on a general $\Omega^{(c)} \setminus \{\lambda\}$ is not available, but Chebyshev [14] and Faber polynomials [15] provide approximate solutions of it.

Since the algorithm starts from a random initial vector, it can happen that some eigenvalues within $\Omega^{(a)}_{D_W}$ are not found in the Arnoldi iteration. Especially, some in a set of closely lying or exactly degenerate eigenvalues might be missing. This effect is considerably reduced by the polynomial transformation.

For an appropriate polynomial, the focusing effect and the acceleration by far compensate the costs of the additional multiplications. Eigenvalues in the original spectrum obtained with a polynomial transformation are shown in Fig. 1.

## III. THE PEELING TRANSFORMATION

In previous investigations of the lowest real eigenvalues of $D_W$ in lattice QCD [13], a certain set of simple polynomials has been proposed. It consists of power transformations of the form

$$p_{\rm power}(D_W; n, \sigma) = (D_W + \sigma {\rm I})^n \,, \text{ with } n \in \mathbf{N},\, \sigma \in \mathbf{R} \,. \tag{2}$$

It has been shown to considerably improve the performance of Arnoldi algorithm. [2] The effect of this transformation on a test eigenspectrum is illustrated in Fig. 2(b). The region of computed eigenvalues in the original spectrum gets a wedge like shape. Hence, the computation is better focused on $\Omega^{(f)}_{D_W}$. However, at larger $n$ the focusing effect saturates.

Based on these observations, we propose here the "peeling transformation" as an iteration of the power transformation. It consists in the following steps:

1. The starting point is a power transformation with an additional renormalization factor $r_0 \in {\rm R}$, $p_0(D_W; n_0, \sigma_0, r_0) = p_{\rm power}(D_W/r_0; n_0, \sigma_0)$.

2. For the resulting eigenspectrum, a new power transformation is chosen for a further focusing on $\Omega^{(f)}_{D_W}$,
   $p_1(D_W; n_0, \sigma_0, r_0, n_1, \sigma_1, r_1) = p_{\rm power}\left(p_0(D_W; n_0, \sigma_0, r_0)/r_1; n_1, \sigma_1\right))$.

3. This procedure is iterated until the polynomial $p_N$ is obtained.

[2] A choice of parameters $n$ and $\sigma$ is explained in Sec. VI.

The effect of the further iterations on a test eigenspectrum is shown in Fig. 2. Clearly, the eigenvalues in $\Omega^{(f)}_{D_W}$ are made accessible by the transformation, while $\Omega^{(c)}_{D_W} \setminus \Omega^{(f)}_{D_W}$ is compressed in a region close to the transformed zero. The polynomial resulting from the iteration is

$$p_N(D_W; n_0, \sigma_0, r_0, \ldots, n_N, \sigma_N, r_N) = (\cdots((D_W/r_0 + \sigma_0 \mathrm{I})^{n_0} + \ldots)/r_N + \sigma_N)^{n_N}\,, \qquad (3)$$

with the free parameter $n_i$, $\sigma_i$, and $r_i$.[3] The question of an optimal choice for these parameters depends on the form of the eigenspectrum and is addressed in Sec. VI.

## IV. REAL EIGENVALUES AND DETERMINANT SIGNS

One of the goals of our calculations were the determinant signs for numerical simulations of one-flavor QCD and Pfaffian signs for the supersymmetric Yang-Mills theory. In order to realize small pion or gluino masses, both theories were simulated within a parameter regime where very small and negative eigenvalues appear. Except for the real modes, all eigenvalues of $D_W$ appear in complex conjugate pairs. Thus the determinant and Pfaffian signs depend only on the real negative eigenvalues; in particular

$$\det(D_W) = \prod_{\{\lambda \in \mathbf{C}\,|\,\Im[\lambda]>0\}} |\,\lambda\,|^2 \prod_{\Im[\lambda]=0} \lambda, \qquad (4)$$

see [17, 18] for more details. For one-flavor QCD, Fig. 3 illustrates the distribution of the lowest eigenvalues for two different $\kappa$ values.

This method, based on a direct computation of the real negative eigenvalues of $D_W$, turns out to be more efficient than the previously considered "eigenflow" approach, where determination of the sign is based on the eigenvalues of $\gamma_5 D_W$. This Hermitian matrix allows to compute its eigenvalues by means of simpler computational methods. However, in order to obtain the determinant (Pfaffian) sign the eigenvalues have to be computed at several different $\kappa$ values [17, 18].[4]

Depending on the parameters of the simulation, in particular on the value of the hopping parameter $\kappa$, we obtained up to 50% negative signs in the simulation of one-flavor QCD. With increasing $\kappa$, approaching its critical value $\kappa_c$, the number of negative signs increases. For values of $\kappa$, which have been used for measurements of the particle spectrum, however, the number of negative signs was well below 10%.

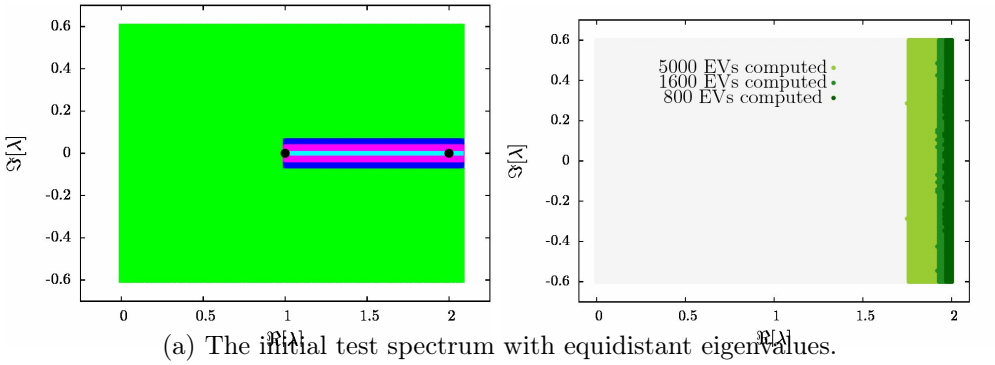
## V. COMPARISON OF THE POLYNOMIAL TRANSFORMATIONS

To demonstrate the importance of the different steps in the peeling method, we compare the performance of several different peeling polynomials with power polynomials. The computation time needed to obtain a number of wanted eigenvalues allows for a simple and clear representation of this performance. In the present case, $\Omega^{(f)}_{D_W}$ is the region of all eigenvalues with an imaginary part whose absolute value is smaller than 0.05. The polynomials were constructed as described in Section VI.

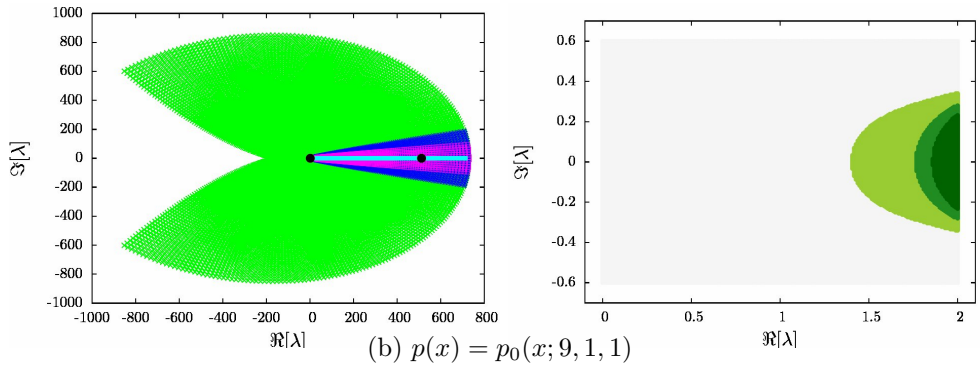
---

[3] The parameter $r_i$ can be absorbed into a redefinition of the $\sigma_i$ and an overall rescaling.

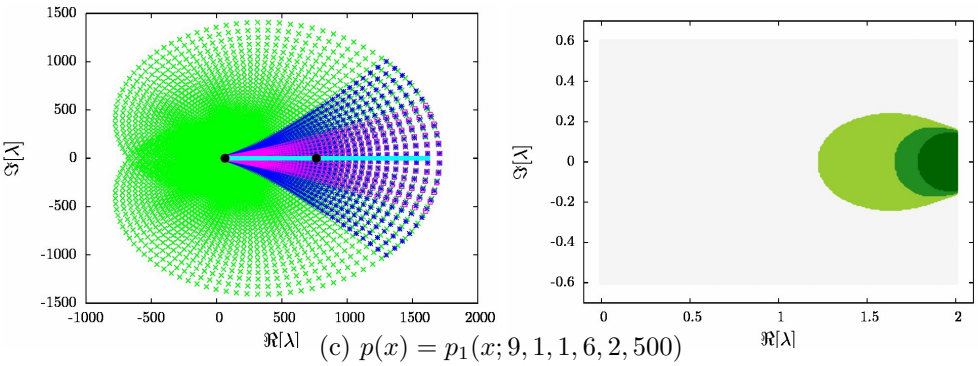
[4] This method is similar to the computation of all real eigenvalues in [19].



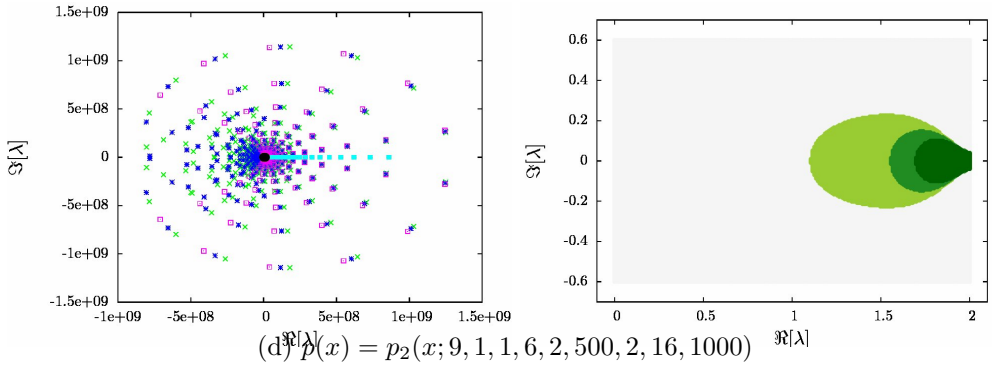

(a) The initial test spectrum with equidistant eigenvalues.




(b) $p(x) = p_0(x; 9, 1, 1)$




(c) $p(x) = p_1(x; 9, 1, 1, 6, 2, 500)$




(d) $p(x) = p_2(x; 9, 1, 1, 6, 2, 500, 2, 16, 1000)$

FIG. 2. The figures on the left hand side show the result of the polynomial transformation applied to a test spectrum of equidistant eigenvalues that fill a rectangular region. The small region colored in light blue corresponds to $\Omega^{(f)}_{D_W}$. The figures on the right display the parts in the original spectrum computed successively in the polynomial Arnoldi iterations. In the first figures no polynomial is applied. The polynomials applied in the other figures are listed below each of them.

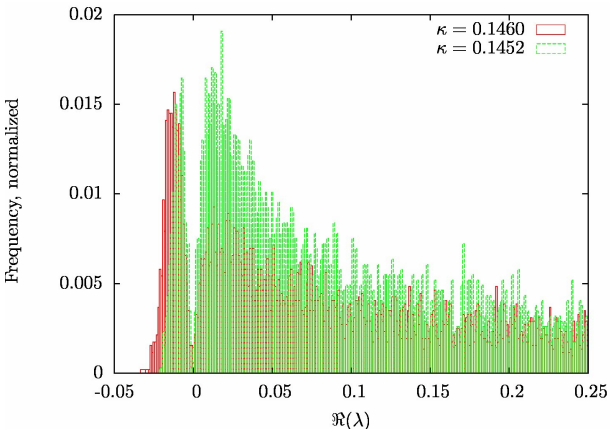



FIG. 3. Distributions of the lowest real eigenvalues of the even-odd preconditioned Wilson-Dirac operator in simulations of one-flavor QCD. When $\kappa$ is increased, the distribution is shifted and a larger peak in the negative sector is observed.

The eigenvalue computations depend on the number of computed eigenvalues, the size of the available eigenspace, and the maximal number of iterations. We have varied all of these parameters for the comparison of the polynomials. Fig. 4 shows the performance of different orders of the power method compared to peeling polynomials of a similar order. Clearly the peeling polynomials allow for a more efficient calculation of the eigenvalues in the considered region. The improvement of the Arnoldi extraction by the polynomials seems to be saturated at a certain order. The extra number of matrix vector multiplications compensates the focusing and acceleration effect. This saturation happens at higher orders for the peeling polynomials than for the power polynomials. In case of the peeling polynomials, the saturation depends on the eigenvalue density, since for a larger lattice size it happens at a larger order. At a smaller lattice size the eigenvalue density seems to be too low to profit from the better focused calculation. Eventually the performance is limited when the next region of a high eigenvalue density is reached. In the considered spectra these regions form a regular pattern similar to the free theory (cf. Fig. 1). Therefore, the limiting high eigenvalue density can be attributed to the first doublers. A steep rise of computation time is visible at this point, especially on larger lattices.

Besides the time of the computation the required memory can be a limitation of the eigenvalue computations. In that respect, the peeling approach exhibits decisive advantages with respect to the power method. This is shown in the right part of Fig. 5, where the needed memory is represented by the number of vectors used in the computation. With respect to this requirement even quite large orders of the peeling polynomials can lead to an improvement.

Using larger orders of the peeling polynomials on smaller lattices one observes that most of the time for the Arnoldi computation is spent on a certain set of configurations with a low eigenvalue density inside $\Omega^{(f)}_{D_W}$. To avoid this effect one can imply a small limit on the maximal number of iterations such that a smaller number of eigenvalues is extracted on these rather uninteresting configurations. The resulting improvement is shown in the right part of Fig. 5. Nevertheless, one should be careful with the limit on the maximum number of iterations. The chance of missing some eigenvalues is increased when the number of iterations gets very low. The Arnoldi algorithm requires a balance of the number of

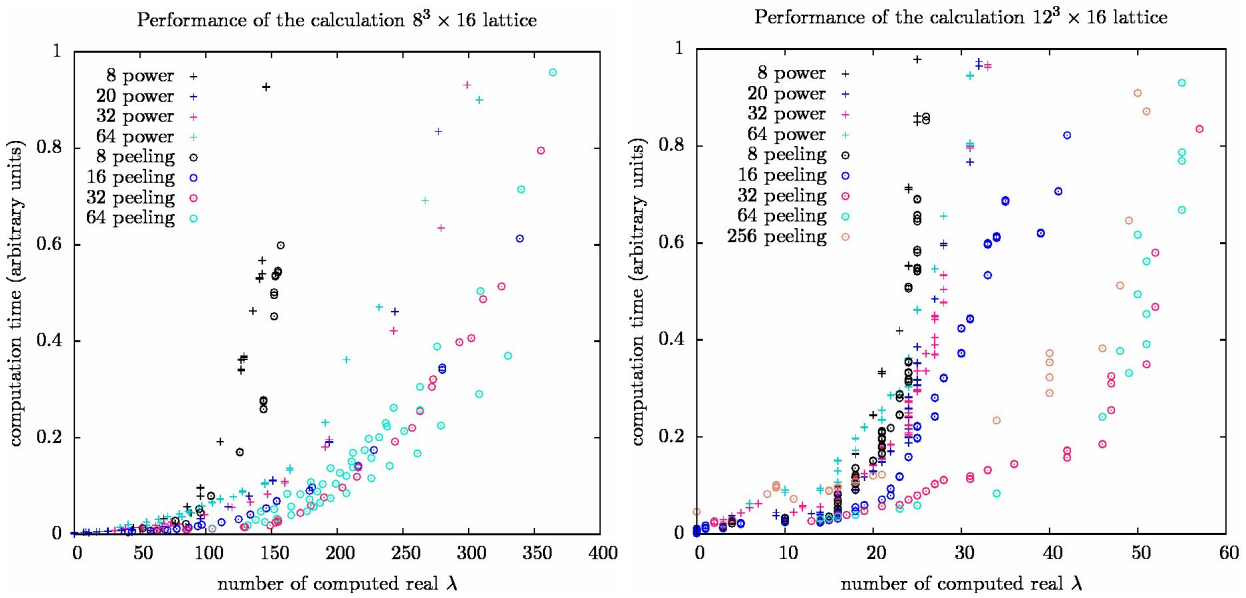



FIG. 4. The performance of the different polynomials with different sizes of the eigenspace and different maximal number of iterations. The numbers in front of the polynomial name indicate the polynomial order. The computations were done on 91 / 11 configurations of one-flavor QCD for the $8^3 \times 16$ / $12^3 \times 16$ lattice. All eigenvalues counted for the number of real $\lambda$ have an imaginary part smaller than 0.05. The large variation for the order 64 peeling polynomial on the small lattice shows a dependence on the maximal number of iterations as in Fig. 5.

multiplications in the polynomial and of the Arnoldi iterations.

## VI. TECHNICAL DETAILS OF THE POLYNOMIAL DESIGN

The polynomials can be designed to improve the focusing on the largest or the smallest real eigenvalues. To simplify the notation, we assume that in the latter case the transformation $D_W \to 2 - D_W$ is applied, such that again the largest real eigenvalues should be computed.[5] The region of the wanted eigenvalues is hence $\Omega^{(f)}_{D_W} = \{\lambda_i | \, |\Im[\lambda_i]| < \varepsilon, \Re[\lambda_i] \geq x_{\min}\}$, where $\varepsilon > 0$ is small and $x_{\min}$ is the deepest point in the spectrum considered in the computation. For simplicity, the normalization of the polynomial is chosen such that $p(x_{\min}) = 1$ in each step of the peeling transformation. Thus, $\sigma_k$ is replaced using $\sigma_0 = 1 - x_{\min}/r_0$ or $\sigma_k = 1 - 1/r_k$ for $k > 0$.

To understand the effect of one step of the peeling transformation (i. e. a power transformation), we represent the complex eigenvalues $\lambda_i$ by their radius and phase,

$$\rho_i = \sqrt{(\Re[\lambda_i]/r_0 - \sigma_0)^2 + (\Im[\lambda_i]/r_0)^2} \text{ and } \theta_i = \arctan\left[(\Re[\lambda_i] - \sigma_0 r_0)/\Im[\lambda_i]\right] ,$$

after the shift and rescaling. The phase is mapped onto $n_0\theta_i$ and a fraction of the eigenvalues with a nonzero imaginary part are hence "rotated away" from the real axis and out of the region of the computed eigenvalues. This effect focuses the calculation on the real eigenmodes. However, the eigenvalues with the largest $\theta$ can be "rotated" inside the computed

[5] For $D_W$, this is a symmetry transformation and the spectrum remains unchanged, but we applied the method also for other operators, like the even-odd preconditioned Wilson-Dirac operator.

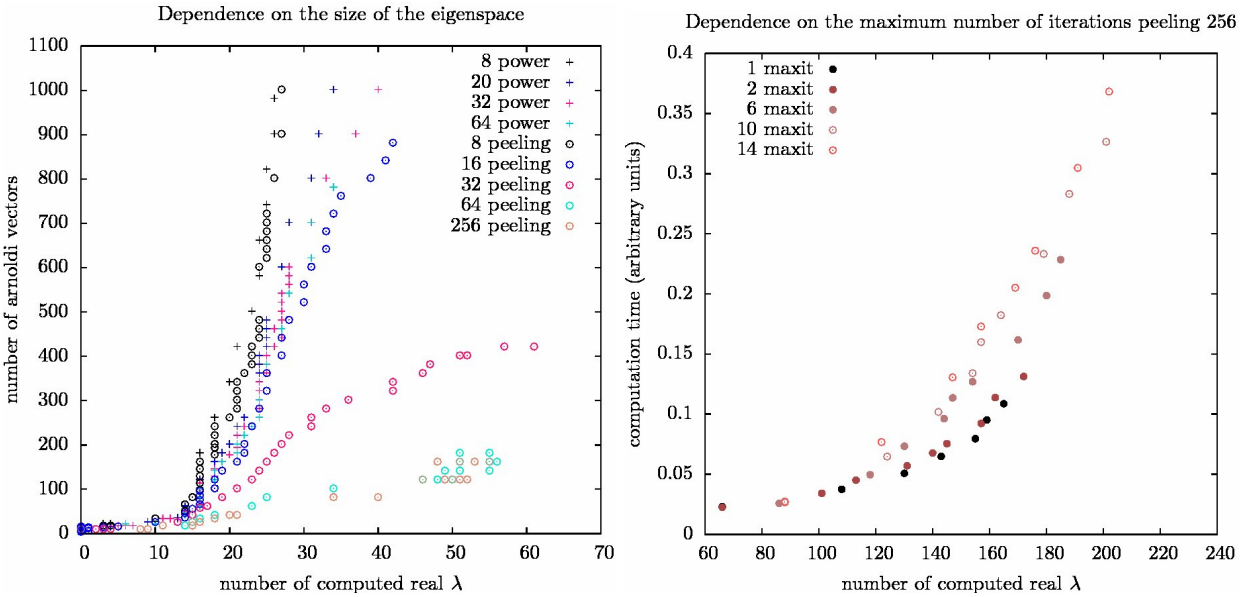



FIG. 5. The figure on the left hand side illustrates the memory (number of vectors) required for the computation of a number of wanted eigenvalues ($12^3 \times 16$ lattice). The eigenspace size was set to twice the number of considered eigenvalues plus two. On the right, we show the dependence on the maximal number of iterations allowed in the Arnoldi algorithm for an order 256 peeling polynomial ($8^3 \times 16$ lattice). In this figure the number of considered eigenvalues is enlarged, but the maximum number of iterations is kept at a small value.

region. Let $\theta_{\max}$ be the maximal phase of all $\lambda_i$ with $\rho_i \geq 1$. One way to avoid such an entering of the "rotated" eigenvalues is to apply the restriction $n_0\theta_{\max} < 3\pi/2$.

The focusing effect can be better controlled when it is visualized by a plot of the contour $\Re\left[p(\lambda)\right] = 1$ in the complex plane. The eigenvalues in the region of all $\lambda$ with $\Re\left[p(\lambda)\right] \geq 1$, i. e. inside the contour, are computed by the algorithm when it reaches the real eigenvalue $\lambda = x_{\min}$. There are $n_0$ of such regions, and the contours surrounding them tend for $\rho \to \pm\infty$ to the lines $\theta = (2l+1)\pi/(2n_0)$, with $l = 0, 1, \ldots, n_0-1$. The larger the number of eigenvalues in $\Omega^{(f)}_{D_W}$ divided by the number of eigenvalues in these regions, the better is the focusing of the polynomial. The restriction to avoid an entering of the "rotated" points in the computed region can now be made more precise: the parameters are restricted such that only the region of one contour surrounding $\Omega^{(f)}_{D_W}$ has overlap with $\Omega^{(c)}_{D_W}$. This region should be made as small as possible for the best focusing. Thus, for a given $n_0$, $r_0$ must be minimized as much as possible without the appearance of a second contour inside $\Omega^{(c)}_{D_W}$.[6]

Increasing $n_0$ and adjusting $r_0$ by this minimization, one observes that the improvement of the focusing saturates at larger $n_0$. The contour lines become almost parallel equidistant lines for large $\rho$ (see first plot in Fig. 6).

The optimization of the focusing can be applied in each step of the peeling strategy: one has to choose a power $n_k$ and minimize $r_k$. The polynomial that deviates most from the power polynomial of the same order has $n_k = 2$ for all $k$. It is shown in the second plot of Fig. 6. Compared to the optimal power polynomial of the same order, the region $\{\lambda | \Re\left[p(\lambda)\right] \geq 1\}$ is narrower at the parts of the spectrum that are computed first and slightly

[6] The minimization of $r_0$ leads to a maximal $\theta$ and hence to a maximal effect of the "rotation".

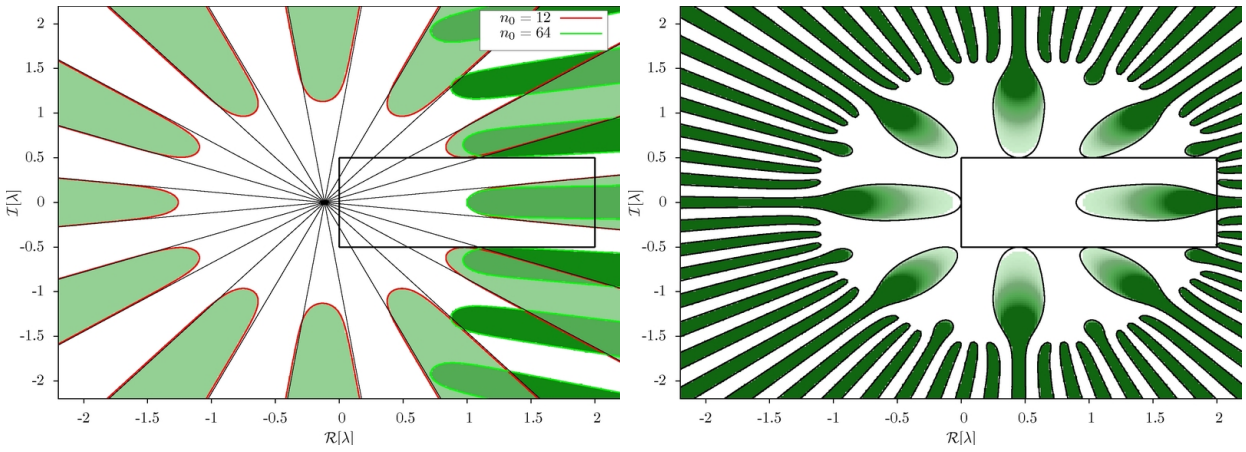



FIG. 6. The first plot shows the region $\{\lambda|\Re\,[p(\lambda)] \geq 1\}$ and the surrounding contours for two power polynomials obtained in the optimization procedure ($x_{\min} = 1$). The black rectangle shows the assumed region $\Omega^{(c)}_{D_W}$ and the gray lines are $\theta = (2l+1)\pi/(2n_0)$ with $l = 0, 1, \ldots, n_0 - 1$ for $n_0 = 12$. The second plot shows the result of an iterated optimization of the peeling strategy using $n_k = 2$ for $k = 0, \ldots, 5$ (overall order of the polynomial 64). The eigenvalues in the regions with the darkest green are computed first.

broader for the inner parts of the spectrum. Keeping the overall order of the polynomial (product of the $n_k$) fixed, one can adjust the region inside the contour. A larger $n_0$, for example, leads to a narrowing of the contour in the inner parts of the spectrum and a broadening in the outer parts. For the comparison in Section V we have chosen $n_k = 2$ for all $k$. This seems to be the best choice for the eigenvalues of $D_W$ in the outer part of the spectrum. In practice it is profitable to test several different polynomials.

In practical applications some choices of the polynomial might severely lower the precision in the multiplication. It might, therefore, be necessary to adapt the parameters, the normalization, and the representation of the polynomial. This problem occurs in particular for high orders of the polynomials.

We have calculated the eigenvalues and eigenvectors of the transformed operators $p(D_W)$ using the restarted Arnoldi algorithm provided by the ARPACK package [6]. The eigenvalues of $D_W$ were obtained from the eigenvectors. Note that in several calculations we have used the even-odd preconditioned Wilson-Dirac operator instead of $D_W$. The eigenvalues of the preconditioned operator $\lambda_i^{(p)}$ are obtained from the eigenvalues of $D_W$ using $\lambda_i^{(p)} = 2\lambda_i - \lambda_i^2$. In the region of interest this relation is invertible.

## VII. CONCLUSION

The polynomials obtained with the peeling strategy lead to an efficient calculation of the smallest (or largest) real eigenvalues of the Wilson-Dirac operator with the Arnoldi algorithm. As we have shown in this work they are better adapted for the eigenvalue distribution of this operator than simple power transformations. The efficiency of the peeling strategy has two main reasons: it circumvents the saturation of the focusing effect in the power transformation and the narrowing of the computed region in the outer parts of the

spectrum avoids a calculation of regions with a large eigenvalue density close to the real axis. Besides this better focusing effect, it provides also an acceleration of the Arnoldi algorithm. We have presented a concrete procedure for the optimization of the parameters of the polynomials in Section VI.

We have also tested Faber polynomials [15] for the computation of the lowest eigenvalues. They offer an interesting alternative with a similar performance as the peeling polynomials in the outer parts of the spectrum. A detailed comparison will be the subject of future work.

The procedure might be adapted for the eigenvalue distribution of other operators with a spectrum in a connected region of the complex plane.

## ACKNOWLEDGMENTS

We thank Federico Farchioni, Istvan Montvay, Gernot Münster, Umut Özugurel and Urs Wenger for helpful comments and discussions. This work was supported by the German Science Foundation (DFG) under contracts Mu 757/13-2 and Mu 757/16-1, and by the John von Neumann Institute of Computing (NIC) with grants of computing time.

---

[1] F. Farchioni, I. Montvay, G. Münster, E. E. Scholz, T. Sudmann and J. Wuilloud, Eur. Phys. J. C **52** (2007) 305 [arXiv:0706.1131 [hep-lat]].

[2] K. Demmouche, F. Farchioni, A. Ferling, I. Montvay, G. Münster, E. E. Scholz and J. Wuilloud, Eur. Phys. J. C **69** (2010) 147 [arXiv:1003.2073 [hep-lat]].

[3] M. Creutz, [arXiv:hep-lat/0511052].

[4] H. Neff, N. Eicker, T. Lippert, J. W. Negele and K. Schilling, Phys. Rev. D **64** (2001) 114509 [arXiv:hep-lat/0106016].

[5] D. Darnell, R. B. Morgan and W. Wilcox, Nucl. Phys. Proc. Suppl. **129** (2004) 856 [arXiv:hep-lat/0309068].

[6] ARPACK SOFTWARE, http://www.caam.rice.edu/software/ARPACK/

[7] I. Hip, T. Lippert, H. Neff, K. Schilling and W. Schroers, Nucl. Phys. Proc. Suppl. **106** (2002) 1004 [arXiv:hep-lat/0110155].

[8] F. Bruckmann, C. Gattringer and C. Hagen, Phys. Lett. B **647** (2007) 56 [arXiv:hep-lat/0612020].

[9] F. Synatschke, A. Wipf and C. Wozar, Phys. Rev. D **75** (2007) 114003 [arXiv:hep-lat/0703018].

[10] C. Gattringer and I. Hip, Nucl. Phys. B **541** (1999) 305 [arXiv:hep-lat/9806032].

[11] C. Gattringer and S. Solbrig, Phys. Lett. B **621** (2005) 195 [arXiv:hep-lat/0503004].

[12] J. Wuilloud, PHD thesis, University of Münster, 2010.

[13] H. Neff, Nucl. Phys. Proc. Suppl. **106** (2002) 1055 [arXiv:hep-lat/0110076].

[14] Y. Saad, *Numerical methods for large eigenvalue problems*, Manchester University Press, 1992.

[15] V. Heuveline and M. Sadkane, Elec. Trans. Numer. Anal. **5** (1997) 62.

[16] C. A. Beattie, M. Embree and D. C. Sorensen, SIAM Rev. **47** (2005) 492.

[17] I. Montvay, Int. J. Mod. Phys. A **17** (2002) 2377 [arXiv:hep-lat/0112007].

[18] I. Campos *et al.* [DESY-Münster Collaboration], Eur. Phys. J. C **11** (1999) 507 [arXiv:hep-lat/9903014].

[19] S. Itoh, Y. Iwasaki and T. Yoshie, Phys. Lett. B **184** (1987) 375.