

# Low-lying eigenmodes of the overlap operator in quenched QED

Christian Schmid

supervised by Dr. Martin Gürtler, Prof. Dr. Andreas Schäfer

October 17, 2007

**Abstract:** This is the documentation of a project performed at the chair of Prof. Schäfer in Regensburg. The aims were the calculation of the lowest eigenvalues of the QED overlap operator on a  $12^4$  lattice in quenched approximation close to the phase transition at  $\beta \approx 1.01$  and the analysis of the eigenvalues and eigenvectors with statistical methods. The gauge field was created by a Monte-Carlo algorithm, whereas the eigenvalues were determined using an algorithm based on the Arnoldi method.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Lattice simulations . . . . .	3
1.2	Dirac-Operator . . . . .	4
1.2.1	Wilson fermions . . . . .	4
1.2.2	Overlap fermions . . . . .	5
1.3	Arnoldi method . . . . .	6
<b>2</b>	<b>Implementation</b>	<b>7</b>
2.1	Gauge configurations . . . . .	7
2.2	Overlap package . . . . .	8
2.3	Test-runs . . . . .	10
2.3.1	Embedding U(1) configurations into $3 \times 3$ -matrices . . . . .	10
2.3.2	Plaquette . . . . .	12
2.3.3	Orthogonal subspaces . . . . .	12
2.3.4	Check eigenvectors . . . . .	13
2.3.5	Free case . . . . .	13
2.3.6	Ginsparg-Wilson Circle . . . . .	14
2.3.7	Computation time . . . . .	17
2.3.8	Number of Wilson eigenvectors . . . . .	18
<b>3</b>	<b>Lowest eigenvalues of the overlap operator on a <math>12^4</math> lattice</b>	<b>18</b>
3.1	Zero-modes . . . . .	19
3.2	IPRs . . . . .	20
3.3	Histograms . . . . .	22
3.4	Eigenmodes . . . . .	24
<b>4</b>	<b>Summary</b>	<b>24</b>
<b>5</b>	<b>Appendix: Input parameters</b>	<b>26</b>

# 1 Introduction

Lattice simulations are a non-perturbative approach to quantum field theory. Especially in QCD, where the coupling constant  $\alpha_s$  is of the order of 1, they allow us to determine numerical results for calculations, which cannot be performed analytically because of a non-convergent perturbation series. In QED lattice calculations may mainly serve as a possibility to verify analytical results or as a playground for beginners in lattice QCD. But there are also some problems in QED that can only be handled by lattice simulations. Additionally the lattice can serve as a regularization procedure of UV divergencies, because it generates an upper limit for possible momenta.

With this background I have done my first steps in this interesting wide field by performing some basic simulations with Monte-Carlo methods and calculating the eigenvalues of the overlap Dirac operator. In this document you will find a short introduction to the theoretical background (if you want to go into details, I would suggest to read an introductory paper like [GU]), an overview of the practical work I have done for the calculations including some important tests and finally the analysis of the results.

## 1.1 Lattice simulations

QED lattice simulations are based on the path integral formalism, which allows us to calculate the expectation value of an arbitrary observable in the following way:

$$\langle O \rangle = \frac{1}{Z} \int D [\Psi(x), \bar{\Psi}(x), A_\mu(x)] O [\Psi(x), \bar{\Psi}(x), A_\mu(x)] e^{iS[\Psi(x), \bar{\Psi}(x), A_\mu(x)]} \quad (1)$$

with  $Z = \int D [\Psi(x), \bar{\Psi}(x), A_\mu(x)] e^{iS[\Psi(x), \bar{\Psi}(x), A_\mu(x)]}$  and the QED action

$$S = \int d^4x \bar{\Psi} \underbrace{[i\gamma^\mu \partial_\mu - e\gamma^\mu A_\mu(x) - m]}_K \Psi + \frac{1}{4} F_{\mu\nu} F^{\mu\nu} \quad (2)$$

$K$  is the well-known Dirac operator.

In Euclidian space we perform a Wick rotation  $x_0 \rightarrow -ix_4$ , which replaces  $e^{iS} \rightarrow e^{-S^E}$ , where

$$S^E = \int d^4x \bar{\Psi} [D^E + m] \Psi + \frac{1}{4} F_{\mu\nu} F^{\mu\nu} \quad (3)$$

Additionally we have to discretize the space-time, because computers can only handle a finite number of points. For convenience one usually substitutes the gauge field  $A_\mu(x)$  by so-called link variables  $U_\mu(x)$ , which are defined as:

$$U_\mu(x) = e^{iaeA_\mu(x)} \quad (4)$$

where  $a$  is the lattice spacing and  $e$  the positive elementary charge.

A local gauge transformation  $V(x)$  has the following effect on the fields and link variables (see [GU] p. 27 f.):

$$\begin{aligned} \Psi(x) &\rightarrow V(x)\Psi(x) \\ \bar{\Psi}(x) &\rightarrow \bar{\Psi}(x)V^\dagger(x) \\ U_\mu(x) &\rightarrow V(x)U_\mu(x)V^\dagger(x + \hat{\mu}) \end{aligned} \quad (5)$$

As we have no continuous space-time on the lattice, we have to substitute the derivatives by differences. A convenient choice in terms of errors caused by the lattice spacing  $a$  is the symmetrized difference (cf. [GU] p. 18):

$$\partial_x \Psi(x) \rightarrow \frac{1}{2a} (\Psi(x+a) - \Psi(x-a)) \quad (6)$$

We will need these considerations in the following section, when we construct the discretized Wilson action on the lattice.

For completeness: one often makes use of an important simplification by performing the path integral only over the gauge field  $A_\mu$  leaving  $\Psi$  and  $\bar{\Psi}$  unconsidered. This approach is called "quenched approximation" and avoids immense computational costs.

There are different ways to implement the discretization as discussed in [GU]. One possible solution, the Wilson action, is introduced in the following section 1.2.1. After the discretization we can apply Monte-Carlo simulations to calculate the expectation value of a wide spectrum of observables (1). These methods are well known from statistical physics. You can find more details about that in [KO] or any introduction to lattice QCD like [GU].

## 1.2 Dirac-Operator

In the following you will find two important realizations of the Dirac operator on the discrete space-time lattice.

### 1.2.1 Wilson fermions

For the discretization on the lattice we have to substitute the derivatives by differences (see (6)) and add appropriate gauge links to maintain gauge invariance (see (5)). So a first naive approach is to transform the continuous action in equation (2) without the gauge field part  $\frac{1}{4}F_{\mu\nu}F^{\mu\nu}$  to the following gauge formulation (see [GU] p. 30 ff.):

$$S^N = m \sum_n \bar{\Psi}(n)\Psi(n) + \frac{1}{2a} \sum_n \bar{\Psi}(n)\gamma_\mu \left[ U_\mu(n)\Psi(n+\hat{\mu}) - U_\mu^\dagger(n-\hat{\mu})\Psi(n-\hat{\mu}) \right] \quad (7)$$

The problem of this naive discretization of the Dirac action is the so-called Fermion Doubling (see [GU] p. 41 f.). To avoid this trouble Wilson introduced an additional

irrelevant operator. The Wilson action can then be formulated as ([GU]):

$$\begin{aligned}
S^W &= m \sum_n \bar{\Psi}(n) \Psi(n) \\
&+ \frac{1}{2a} \sum_{n,\mu} \bar{\Psi}(n) \gamma_\mu \left[ U_\mu(n) \Psi(n + \hat{\mu}) - U_\mu^\dagger(n - \hat{\mu}) \Psi(n - \hat{\mu}) \right] \\
&- \frac{r}{2a} \sum_{n,\mu} \bar{\Psi}(n) \left[ U_\mu(n) \Psi(n + \hat{\mu}) - 2\Psi(n) + U_\mu^\dagger(n - \hat{\mu}) \Psi(n - \hat{\mu}) \right] \\
&= \frac{ma + 4r}{a} \sum_n \bar{\Psi}(n) \Psi(n) \\
&+ \frac{1}{2a} \sum_n \bar{\Psi}(n) \left[ (\gamma_\mu - r) U_\mu(n) \Psi(n + \hat{\mu}) - (\gamma_\mu + r) U_\mu^\dagger(n - \hat{\mu}) \Psi(n - \hat{\mu}) \right] \\
&= \sum_{n,m} \bar{\Psi}^L(n) M_{nm}^W \Psi^L(m)
\end{aligned} \tag{8}$$

with the interaction matrix

$$M_{nm}^W a = \delta_{n,m} - \kappa \sum_\mu \left[ (r - \gamma_\mu) U_\mu(n) \delta_{n,m-\hat{\mu}} + (r + \gamma_\mu) U_\mu^\dagger(n - \hat{\mu}) \delta_{n,m+\hat{\mu}} \right] \tag{9}$$

and the rescaled terms

$$\begin{aligned}
\kappa &= \frac{1}{2ma + 8r} \\
\Psi^L &= \sqrt{ma + 4r} \Psi = \frac{\Psi}{\sqrt{2\kappa}}
\end{aligned} \tag{10}$$

(e.g. see [GU] p. 46). The Wilson parameter is set to  $r = 1$ .

One big disadvantage of the Wilson action is the fact, that it breaks chiral symmetry.

### 1.2.2 Overlap fermions

Overlap fermions have an exact chiral symmetry on the lattice, which avoids additive renormalization of the quark masses ([LU]). The overlap operator, we want to examine in this project, is defined as ([TA], [IA], [NE1], [NE2]):

$$D \equiv \rho \left[ 1 + \gamma_5 \frac{H_W}{\sqrt{H_W^2}} \right] = \rho [1 + \gamma_5 \text{sgn}(H_W)] \tag{11}$$

where  $H_W$  is the hermitian Wilson Dirac operator  $H_W = \gamma_5 D_W$  ( $D_W$  is the Wilson Dirac operator), and the negative mass  $\rho \in (0, 2)$ . The overlap operator fullfills the Ginsparg-Wilson identity (see section 2.3.6).

Apart from the conservation of chiral symmetry a related aspect of the overlap operator is the possibility to determine the topological charge via the index theorem ([HA], [IA],

[BE]), as we do in section 3.1. However, the calculation of the overlap operator consumes much more CPU power than for the Wilson operator.

For numerical calculations the *sgn*-function on right hand side of (11) is computed as ([IA], [NE3]):

$$\text{sgn}(H_W) = \sum_{i=1}^N \text{sgn}(\lambda_i) \Psi_i \Psi_i^\dagger + P_\perp^N \text{sgn}(H_W) \quad (12)$$

where  $\Psi_i$  are known eigenvectors of the Wilson Dirac operator and  $\lambda_i$  the according eigenvalues. The first part of the right hand side of (12) is treated exactly, whereas the second part is approximated by a polynomial. The degree of this polynomial is chosen proportional to the condition number  $\kappa_c$ , which is defined as the squared ration of the biggest eigenvalue of the Wilson operator to the  $N + 1$ th lowest Wilson eigenvalue ([IA]):

$$\kappa_c = \frac{\mu_{max}^2}{\mu_{N+1}^2} \quad (13)$$

So to improve the computation time for the overlap operator by lowering the condition number  $\kappa_c$ , we first determine  $N$  eigenvectors of the Wilson Dirac operator.

### 1.3 Arnoldi method

The program I have used for the numerical calculation of the eigenvalues and eigenvectors of the Dirac operator is based on the Arnoldi method. This algorithm is especially suitable to determine the eigenvalues and eigenvectors of a large  $n \times n$ -matrix with few non-zero entries and some kind of structure, which reduces the costs for a matrix-vector-product ([ARP]). The latter aspect is important, because the algorithm does not access the individual matrix elements directly, but deals with the change of a vector under repeated multiplication with the matrix ([WI], [KR]).

An intuitive approach to obtain eigenvalues of a matrix  $A \in \mathbb{C}^{n \times n}$  numerically is the power iteration. Starting with an arbitrary vector  $v$  the series  $Av, A^2v, A^3v, \dots, A^{m-1}v$  converges to the eigenvector belonging to the dominant eigenvalue  $\lambda_l$  (greatest absolute value), if this eigenvalue is simple and  $v$  is not orthogonal to the according eigenvector.

If we calculate an orthonormal basis of the Krylov-space, defined as

$$K_m(A, v) = \text{span} \{v, Av, A^2v, \dots, A^{m-1}v\} \quad (14)$$

these basis vectors can be used as an approximation of the eigenvectors corresponding to the  $m$  largest eigenvalues of  $A$  ([WI]).

The problem with this method is, that for large  $k \rightarrow \infty$  the vectors  $A^{k-1}v$  and  $A^k v$  are nearly parallel, i.e. linearly dependent. So the calculation of an orthonormal set of basis vectors of  $K_m$  according to Gram-Schmidt is badly conditioned and therefore instable.

The Arnoldi method avoids this problem by doing the orthonormalization of  $v_k$  according to  $\text{span}\{v, v_1, \dots, v_{k-1}\}$  directly after each matrix multiplication  $v_k = Av_{k-1}$ . The resulting vectors are called Arnoldi vectors. Basically they are used to reduce the eigenvalue problem of  $A$  to the determination of eigenvalues of a Hessenberg-matrix, which

can be done e.g. with the QR-algorithm because of the convenient form of this kind of matrix. If you want to know more details or the explicit algorithm, see [KR].

As already mentioned the Arnoldi method is convenient to handle a matrix with some kind of structure and many zero-entries like the Dirac operator. Additionally the algorithm converges to the extremal eigenvalues of a matrix, so it is perfectly applicable to our problem.

## 2 Implementation

For the determination of the eigenvalues of the overlap operator, I used the following existing programs:

- a Monte-Carlo simulation to create the gauge field configurations  $A_\mu(n)$  or link variables  $U_\mu(n)$  (written by Sonja Koller [KO])
- Arnoldi routines from the overlap package to calculate the eigenvalues of the overlap operator for the specific gauge configurations (written by Thomas Streuer [ST])

Especially the overlap package contains many programs, which have been optimized using assembler commands. It was quite convenient to be able to use this package, because numerical routines to calculate the eigenvalues of a huge matrix like the Dirac operator on a  $12^4$  lattice consume a lot of time, and it would have been impossible to develop such a high-performance code on my own in an acceptable amount of time.

To use the former programs for my purpose I in principle had to adapt the provided codes, implement an interface between the different programs, run them with the proper input parameters and finally evaluate the output using some self-developed tools.

### 2.1 Gauge configurations

In order to create the U(1) gauge configurations  $A_\mu(n)$  or link variables  $U_\mu(n)$  on the lattice I extended a Monte-Carlo simulation developed by Sonja Koller in a former project at the chair of Prof. Schäfer ([KO]). Sonja's program generates several gauge configurations using the Metropolis algorithm in order to determine the expectation value of some significant observables namely the plaquette, the free energy and the Wilson loops.

This method is based on path integrals and the importance sampling method according to the Boltzmann distribution in statistical physics (cf. section 1.1). Sonja described this elementary idea for her calculations quite well in the documentation of her project [KO], so I will not discuss the details of that method.

In order to get suitable data for the overlap calculation routine I basically had to modify the three following points in Sonja's program:

- To use memory efficiently Sonja did not work with an array of all link variables  $U_\mu(n)$  but instead used the plaquette values  $U_{\mu\nu}(n)$ .

The plaquette is the smallest possible Wilson loop built of 4 link variables and is defined as:

$$\begin{aligned} U_{\mu\nu}(n) &= U_\mu(n)U_\nu(n + \hat{\mu})U_\mu^\dagger(n + \hat{\nu})U_\nu^\dagger(n) \\ &= \exp\{iae[A_\mu(n) + A_\nu(n + \hat{\mu}) - A_\mu(n + \hat{\nu}) - A_\nu(n)]\} \end{aligned} \quad (15)$$

Although this provided enough data to obtain her desired observables, the information did not suffice for my purpose, because I needed the values of  $U_\mu(n)$  as input data for the routine, which calculates the eigenmodes. So I had to adjust the code in the Monte-Carlo step, which updates the gauge configuration, in such a way, that it uses  $U_\mu(n)$  instead of the plaquette.

- After producing a gauge configuration I had to write the array with the link variables  $U_\mu(n)$  into an output file using the correct format for the eigenvalue algorithm. One of the major problems in this context was, that the overlap package of Thomas Streuer was implemented for QCD calculations, i.e. SU(3) fields. I tried out two different ways to integrate the U(1) data into the SU(3) procedure. (You can find more about this in section 2.3.1.) As it seems to be convenient to combine three U(1) gauge configurations to one  $3 \times 3$  configuration, I had to write the values  $U_\mu^i(n)$  at each lattice point into the same output file, where  $i = 1, 2, 3$  represents three different configurations created by the Monte-Carlo algorithm.

With these modifications I could use Sonja's program to generate gauge configurations as a basis for the calculation of eigenvalues of the overlap operator. As I wanted to perform statistical studies of the lowest eigenvalues on both sides of the phase transition, I created  $\sim 50$  configurations for each value of  $\beta = 0.99$  and  $\beta = 1.03$ .

## 2.2 Overlap package

After generating a gauge configuration with the Monte-Carlo algorithm the main step was the calculation of the eigenvalues of the overlap operator for this specific configuration. This part needs lots of time and computer power, because we are dealing with a matrix which has  $(12N)^2$  entries (3 colors and 4 spinor components), where  $N$  is the number of lattice sites, i. e. in our case  $N = 12^4$ .

Because of the complexity of this problem I didn't implement the numerical calculation of the eigenvalues and eigenvectors on my own, but used a program package developed by Thomas Streuer, which is based on the Arnoldi method. These applications are mainly written in C++ and assembler and are therefore highly optimized for the determination of the eigenvalues for different lattice actions (Wilson, overlap etc.). The most important data required by the programs are the gauge configuration  $U_\mu(x)$  and the parameters, which determine the mass, the number of desired eigenvalues, their accuracy etc. (see [ST]).

Unfortunately there are hardly any comments in the code, so many calculations did not work at first try because of wrong input parameters. I had to invest some time to become familiar with the overlap package, until I could start my actual calculations.



First I worked on small lattices of  $4^4$  and  $8^4$  sites respectively, to reduce the time needed for the different attempts, later calculations were done on a  $12^4$  lattice.

As input parameters the fermion mass and lattice constant were chosen as  $ma = -1.6$ , which according to (10) results in a value of

$$\kappa = \frac{1}{2ma + 8r} = 0.208\bar{3}$$

with  $r = 1$ .

For my purpose I basically had to perform the following steps to calculate the smallest overlap eigenvalues for each gauge configuration created by the Monte-Carlo algorithm of Sonja Koller's simulation:

- Save the gauge configuration in a file combining three U(1) configurations to one  $3 \times 3$  configuration (see section 2.3.1).
- Convert this plain-number file to a suitable format, using the program "conv\_cdv" of the overlap package.
- Calculate the  $N$  lowest Wilson eigenvalues for the gauge configuration using "condition".
- Calculate the lowest overlap eigenvalues using "arnoldi" and the  $N$  formerly determined Wilson eigenvectors.
- Extract the overlap eigenvalues of the output data using "arn\_evec" and convert them to plain number format using "conv\_cdv".

As the eigenvectors of the Wilson operator are used to calculate the overlap operator (see section 1.2.2), it is necessary to run "condition" before calculating the overlap eigenvalues with "arnoldi".

Among the input settings of "condition" there is a parameter for the maximum absolute value of the desired eigenvalues. With this information the program can perform some transformations, before the actual calculation is carried out ([NEF]), which reduces the computation time for the Wilson eigenvectors on the  $12^4$  lattice from initially more than 2 days to  $\sim 1\text{h } 10\text{min}$  (!). Of course I had to perform the calculation without this option at the beginning. After that I could use this advantageous reduction of computation time, as I then knew the magnitude of the eigenvalues approximately.

Finally with the output data of "condition" I could run the "arnoldi"-program to obtain the eigenvalues of the overlap operator. Although using the Wilson eigenvectors for the representation of the *sgn*-function (equation (12) in section 1.2.2) reduces the computation time for the calculation of the overlap operator, this was one of the most expensive parts of my work. In section 2.3.7 you will find more information about the consumed time.

## 2.3 Test-runs

Before I present the overlap eigenvalues on the  $12^4$  lattice, which are the main results of my analysis, I will give you an insight in the most important tests of the calculation routine, as this was a major part of my work.

### 2.3.1 Embedding U(1) configurations into $3 \times 3$ -matrices

The overlap package of Thomas Streuer was developed for QCD calculations, i.e. for SU(3). As my analysis should be done for U(1), I had to find a way to embed the U(1) gauge configurations into  $3 \times 3$ -matrices.

In QCD each quark has three color components, which are represented by Dirac spinors:

$$\Psi = \begin{pmatrix} \Psi_{red} \\ \Psi_{blue} \\ \Psi_{green} \end{pmatrix} \quad (16)$$

where each  $\Psi_i$  with  $i = red, green, blue$  is a Dirac spinor with four components.

The interaction between the quarks is mediated by eight gluons. If we set the coupling constant  $g = 1$ , the coupling term of the gluons is:

$$A_\mu^a \left( \frac{\lambda^a}{2} \right) \quad (17)$$

which is a  $3 \times 3$ -matrix for each  $\mu \in \{0, 1, 2, 3\}$ , where the  $\lambda^a$ 's are the generators of the su(3) Lie algebra in the fundamental representation. When (17) is acting on (16), you can see, that in general quarks with different color can interact via gluons, as the  $3 \times 3$ -matrix in (17) may have non-zero off-diagonal elements.

In QED the coupling of the photon to a charged Dirac particle is simply proportional to  $A_\mu$  and the Dirac particle is represented by only one spinor  $\Psi$ .

In lattice calculations we are not working with  $A_\mu$ , but with the link variables  $U_\mu(n) = e^{i a e A_\mu(n)}$  in the U(1) case. In SU(3) gauge theory the link variables are represented by  $3 \times 3$ -matrices.

I tried out different concepts to embed the U(1) link variables  $U_\mu(n)$  into the  $3 \times 3$ -matrix. The first approach was to identify the gauge field with one single diagonal element of the former matrix  $A_\mu^a \left( \frac{\lambda^a}{2} \right)$ , set the other two diagonal elements to one and the remaining entries to zero:

$$U_\mu^{3 \times 3}(n) = \begin{pmatrix} U_\mu^{U(1)}(n) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (18)$$

When you use this input format for the overlap package, you can expect to get the desired eigenvectors for the U(1) gauge configuration  $A_\mu$  and the eigenvalues for the free case, because of the other two diagonal elements  $U_\mu(n) = 1$  or  $A_\mu(n) = 0 \forall \mu \in \{0, 1, 2, 3\} \forall n$ .

As this matrix has only zero off-diagonal elements, there should be no interaction between the different colors, i. e. the subspace spanned by the eigenvectors belonging to the configuration with  $U_\mu(n) \neq 1$  is expected to be orthogonal to the subspace spanned by the eigenvectors of the free case and should lie completely in the subspace of one color (see 2.3.3). Additionally the eigenvalues belonging to the free case should be twofold degenerate.

I also calculated the eigenvalues for the following input formats:

$$U_\mu^{same}(n) = \begin{pmatrix} U_\mu^{U(1)}(n) & 0 & 0 \\ 0 & U_\mu^{U(1)}(n) & 0 \\ 0 & 0 & U_\mu^{U(1)}(n) \end{pmatrix} \quad (19)$$

with three times the same gauge configuration and

$$U_\mu^{free}(n) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (20)$$

Indeed for these two configurations all eigenvalues were threefold degenerate. Calculations for the lowest eigenvalues also showed, that the set of lowest eigenvalues of the approach (18) is a subset of the set union of the lowest eigenvalues of  $U_\mu^{same}(n)$  and  $U_\mu^{free}(n)$ .

So in principle we can use the ansatz (18) to calculate the desired eigenvalues. We just have to extract the eigenvalues for the gauge configuration  $U_\mu(n) \neq 1$  from the output of the overlap algorithm. This can be done by regarding the components of the corresponding eigenvectors.

To avoid this sorting of eigenvalues we could also use  $U_\mu^{same}(n)$  as input data and consider each resulting, threefold degenerate eigenvalue only once.

In both of the former scenarios we can handle one gauge configuration in each call of the overlap code. As this costs a lot of time, I tried another approach, which combines three different U(1) gauge configurations to one  $3 \times 3$  configuration. The elementary input matrix is given by:

$$U_\mu^{3conf}(n) = \begin{pmatrix} U_\mu^{U(1),1}(n) & 0 & 0 \\ 0 & U_\mu^{U(1),2}(n) & 0 \\ 0 & 0 & U_\mu^{U(1),3}(n) \end{pmatrix} \quad (21)$$

where  $U_\mu^{U(1),i}(n)$  represent three different gauge configurations ( $i = 1, 2, 3$ ).

As discussed above, there is no mixing between the different configurations, as all off-diagonal elements of the matrix are set to zero. This input format reduces the computational costs nearly by a factor of 3, because the computation time for the formats and is approximately the same.

As we only could use very limited computer resources, the reduction of computation time was one of the major problems in this project, so I chose the latter ansatz to transfer the gauge configurations created by the Monte-Carlo algorithm to the overlap programs, which calculate the eigenvalues of the overlap operator.

### 2.3.2 Plaquette

For the generation of the gauge configurations I used the Monte-Carlo simulation written by Sonja Koller (see section 2.1). As I had to modify several points in the original code in order to use it for my purpose, I did some checks to assure, that it still works properly. E. g. I calculated the plaquette (15) close to the phase transition from the confined to the Coulomb phase at the critical coupling  $\beta_c = 1.0111331(21)$ . (Strictly speaking there is no phase transition for the finite lattice.)

In figure 1 you can see the mean plaquette determined with the program "pq" of the overlap package for each three U(1) configurations embedded in one  $3 \times 3$ -matrix and averaged over 48 configurations for each value of  $\beta$ . The data agrees quite well with other measurements like in [TA].

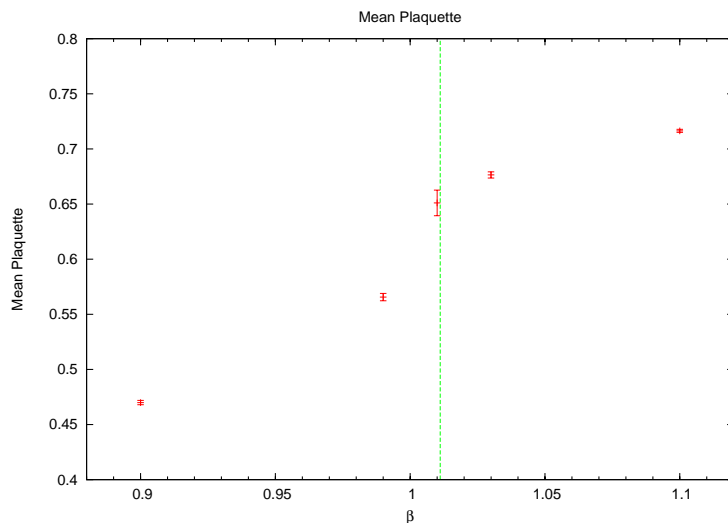


Figure 1: Evolution of the plaquette at the phase transition

So the creation of the gauge configurations and the combination of three configurations to one  $3 \times 3$  configuration seem to work correctly, as the data was obtained from the embedded  $3 \times 3$  gauge configurations.

### 2.3.3 Orthogonal subspaces

As we have discussed in section 2.3.1, there should be no interaction between the three combined gauge configurations. Therefore the eigenvectors belonging to different configurations should span a subspace, which corresponds to exactly one color in the SU(3) space, i.e.

$$\begin{pmatrix} |\Psi_{red}| \\ |\Psi_{blue}| \\ |\Psi_{green}| \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad (22)$$

but may not be a mixed state of two or three different color vectors.

This condition is fulfilled by the eigenvectors resulting from the overlap algorithm with the input data specified in equation (2.3.1), which is another indication that the input format is correct and that the code is working properly.

### 2.3.4 Check eigenvectors

The overlap program package contains an algorithm ("check\_ev") to check, whether the finally calculated eigenvectors are really eigenvectors of the Dirac operator specified by the input data. This algorithm acts with the Dirac operator on the eigenvectors and compares them with the resulting vector.

The results of this test show, that the overlap eigenvectors calculated in this project have the desired accuracy of at least  $10^{-8}$ , which is enough for our purpose.

### 2.3.5 Free case

In section 2.3.1 I mentioned, that I have calculated the eigenvalues for the case of zero gauge field  $A_\mu(n) = 0 \Rightarrow U_\mu(n) = 1$ . There is also an analytical solution for this scenario, which is described e.g. in [WE], p. 55 f.. In that document the calculations are done on a lattice of size  $N = 4^4$ . For that reason and because of the shorter computation time I also chose this lattice size for the free case.

In order to test the eigenvalue algorithm I compared the numerically calculated eigenvalues of the Wilson Dirac operator on a  $4^4$  lattice with the analytical values, which we will obtain in the following.

If we consider the Wilson Dirac operator  $M_{nm}$  (equation (9)) in the case  $U_\mu(n) = 1$  (with  $r = 1$ ), we obtain

$$M_{nm}^{W,free} a = \delta_{n,m} - \underbrace{\kappa \sum_{\mu} [(1 - \gamma_{\mu})\delta_{n,m-\hat{\mu}} + (1 + \gamma_{\mu})\delta_{n,m+\hat{\mu}}]}_{D(n,m)} \quad (23)$$

The eigenvalues of this matrix can be calculated according to [WE] p. 55 f., if we use the Fourier representation of the  $\delta$ -function:

$$\delta_{n,m-\hat{\mu}} = \int_{-\pi}^{\pi} \frac{d^4 p}{(2\pi)^4} e^{ip(n+\hat{\mu}-m)}$$

So  $D(n, m)$  simplifies to:

$$\begin{aligned} D(n, m) &= \sum_{\mu} \int_{-\pi}^{\pi} \frac{d^4 p}{(2\pi)^4} \left[ (e^{ip\hat{\mu}} + e^{-ip\hat{\mu}}) - \gamma_{\mu} (e^{ip\hat{\mu}} - e^{-ip\hat{\mu}}) \right] e^{-ip(n-m)} \\ &= 2 \sum_{\mu} \int_{-\pi}^{\pi} \frac{d^4 p}{(2\pi)^4} [\cos(p_{\mu}) - i\gamma_{\mu} \sin(p_{\mu})] e^{-ip(n-m)} \end{aligned} \quad (24)$$

To obtain the eigenvalue equation, we apply  $D(n, m)$  to a vector  $\Phi e^{-ikm}$ , where  $\Phi$  only depends on the spinor indices:

$$\begin{aligned} \sum_m D(n, m) \Phi e^{-ikm} &= \sum_m \sum_\mu \int_{-\pi}^{\pi} \frac{d^4 p}{(2\pi)^4} [2 \cos(p_\mu) - 2i\gamma_\mu \sin(p_\mu)] \Phi e^{-ipn} e^{-i(k-p)m} \\ &= \sum_\mu [2 \cos(k_\mu) - 2i\gamma_\mu \sin(k_\mu)] \Phi e^{-ikn} \equiv \lambda_D(k_\mu) \Phi e^{-ikn} \end{aligned} \quad (25)$$

Here we have used:

$$\frac{1}{(2\pi)^4} \sum_m e^{-i(k-p)m} = \delta(k-p)$$

To simplify expression (25), consider the Clifford-Algebra of the  $\gamma$ -matrices:

$$\begin{aligned} (a\gamma_\mu + b\gamma_\nu)^2 &= a^2\gamma_\mu^2 + b^2\gamma_\nu^2 + ab \underbrace{\{\gamma_\mu, \gamma_\nu\}}_{\delta_{\mu\nu}} \stackrel{\mu \neq \nu}{=} a^2 + b^2 \\ \left( \sum_\mu a_\mu \gamma_\mu \right)^2 &= \sum_\mu a_\mu^2 \end{aligned}$$

So we obtain:

$$\begin{aligned} \lambda_D(k_\mu) &= \sum_\mu 2 \cos(k_\mu) \pm 2i \sqrt{\left( \sum_\mu \gamma_\mu \sin(k_\mu) \right)^2} \\ &= \sum_\mu 2 \cos(k_\mu) \pm 2i \sqrt{\sum_\mu \sin^2(k_\mu)} \end{aligned} \quad (26)$$

All in all the eigenvalues of (23) are:

$$\lambda(k_\mu) = 1 + 2\kappa \left[ \sum_\mu \cos(k_\mu) \pm i \sqrt{\sum_\mu \sin^2(k_\mu)} \right] \quad (27)$$

with  $k_\mu = \frac{2\pi n_\mu}{s}$ ,  $n_\mu \in \mathbb{N}$  for  $\mu = 1, 2, 3, 4$ .

When you compare the analytically calculated eigenvalues for  $\kappa = 1$  in figure 2 with the numerical solution in figure 3, you will see, that they perfectly agree, up to the fact that numerically not all eigenvalues were calculated.

### 2.3.6 Ginsparg-Wilson Circle

The overlap operator defined in (11) fullfills the Ginsparg-Wilson equation ([WE], [LU], [NE2]):

$$\{\gamma_5, D\} = \gamma_5 D + D \gamma_5 = a D \gamma_5 D \quad (28)$$

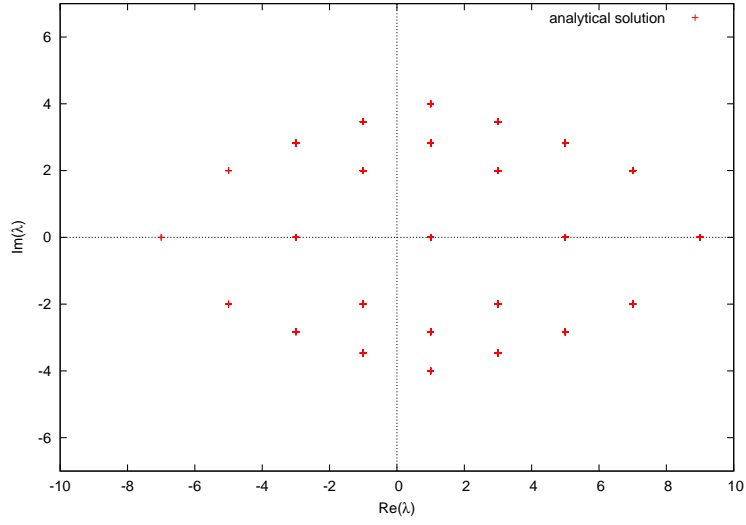


Figure 2: Analytically calculated eigenvalues of the Wilson Dirac operator on a  $4^4$  lattice for the free case  $A_\mu(n) = 0$

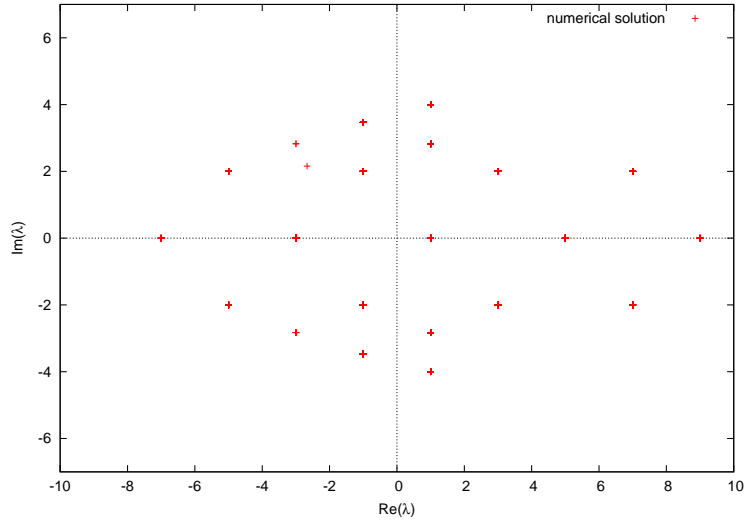


Figure 3: Numerically calculated eigenvalues of the Wilson Dirac operator on a  $4^4$  lattice for the free case  $A_\mu(n) = 0$

From (28) and the  $\gamma_5$ -hermicity of  $D$  we obtain for an eigenvector  $|v\rangle$  of  $D$ :

$$(\gamma_5 D + D \gamma_5) |v\rangle = a D \gamma_5 D |v\rangle = a \gamma_5 D^\dagger D |v\rangle = a \lambda^* \lambda \gamma_5 |v\rangle \quad (29)$$

On the other hand the eigenvalue equation of the overlap operator reads:

$$(\gamma_5 D + D \gamma_5) |v\rangle = (\gamma_5 D + \gamma_5 D^\dagger) |v\rangle = (\lambda + \lambda^*) \gamma_5 |v\rangle \quad (30)$$

These two equations imply:

$$a\lambda^*\lambda = \lambda + \lambda^* \Leftrightarrow |\lambda|^2 = \frac{2}{a}\Re(\lambda) \quad (31)$$

If we choose  $\lambda = x + iy$  with  $x, y \in \mathbb{R}$ , we obtain

$$x^2 + y^2 = \frac{2}{a}x \Rightarrow \left(x - \frac{1}{a}\right)^2 + y^2 = \frac{1}{a^2} \quad (32)$$

which means, that the eigenvalues  $\lambda$  of the overlap operator lie on a circle with radius  $\frac{1}{a}$  and center at  $x = \frac{1}{a}, y = 0$  in the complex plane.

In figure 4 you can see, that the theoretical prediction is fulfilled by the numerically calculated eigenvalues of the overlap operator on a  $4^4$  lattice. This fact asserts the correct implementation of the eigenvalue algorithm.

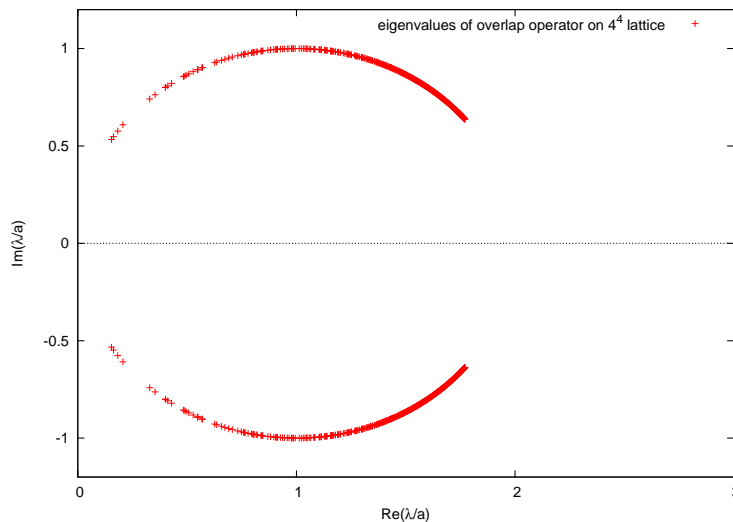


Figure 4: Eigenvalues of the overlap operator for  $\beta = 1.03$  on a  $4^4$  lattice lying on the Ginsparg-Wilson circle in the complex plane

The density of the eigenvalues on the right side of the circle in figure 4 is higher than on the left side. In the case  $a \rightarrow 0$ , the center of the circle would move to the right and the radius would become infinitely large, so that the eigenvalues on the left side of the circle would move towards the imaginary axis.



### 2.3.7 Computation time

For most lattice simulations the available computer time is a major constraint. In my project the most time consuming part was the calculation of the eigenvalues. I spent a lot of effort on reducing these computational costs, because they are the limiting factor for the number of analysed configurations and therefore determine the quantity of our statistical data.

The eigenvalue calculations were mainly performed on an Intel Pentium D with 3.40 GHz and 2 GB memory. So the following numbers refer to the computation time on a comparable computer.

First in figure 5 you can see the time consumed to determine the Wilson eigenvalues using "condition" for different lattice sizes. For these runs I have used the already mentioned reduction of calculation costs by fixing the maximum absolute value of the desired eigenvalues as an input parameter. The polynomial fit shows that the computation time is proportional to  $L^{16/3} = (L^4)^{4/3} = N^{4/3}$ , where  $L$  is the length of the lattice and  $N = L^4$  the volume.

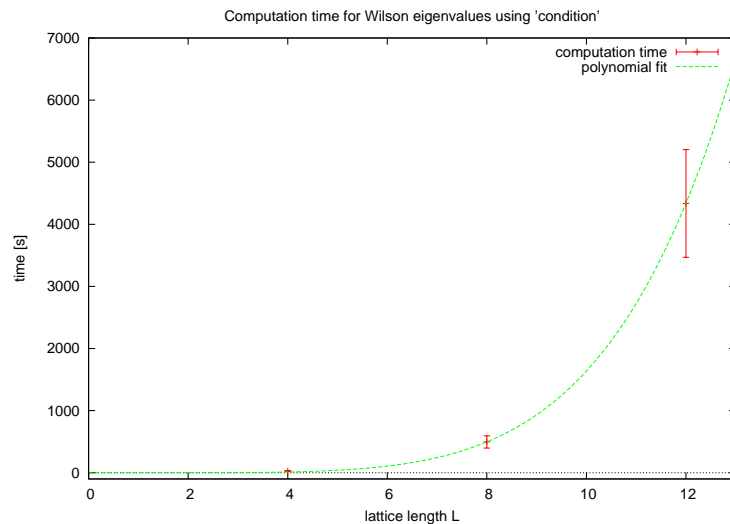


Figure 5: Computation time for the Wilson eigenvalues at different lattice sizes ( $4^4$ ,  $8^4$  and  $12^4$ ) scales as  $t \sim L^{16/3}$

In figure 6 you can see, that the computation time for the overlap eigenvalues is much higher than for the Wilson eigenvalues and furthermore is proportional to  $L^8 = N^2$ . The problem for the overlap operator especially at large lattice size is the limited amount of memory. So the algorithm has to swap data to the hard disc, which costs a lot of additional time.

All in all the reduction of the consumed computation time was a major problem in my project. At the beginning it seemed to be impossible to perform the calculations on an ordinary PC. But after all improvements it is now possible to determine the overlap

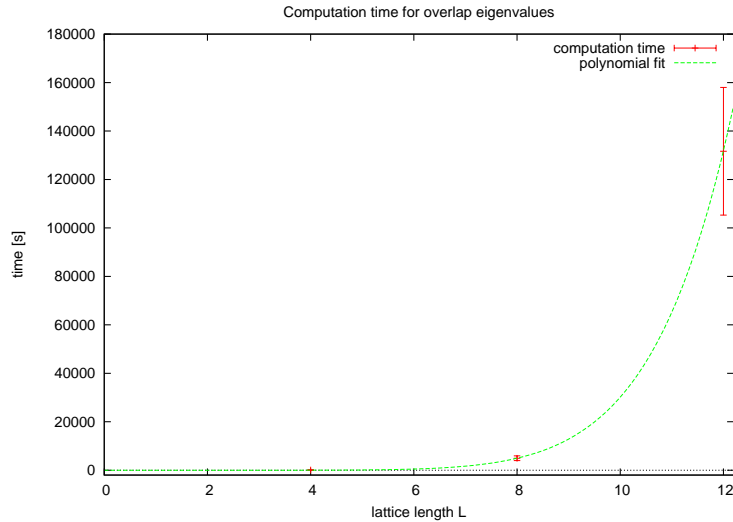


Figure 6: Computation time for the overlap eigenvalues at different lattice sizes ( $N = 4^4$ ,  $8^4$  and  $12^4$ ) scales as  $t \sim L^8$

eigenvalues of all  $2 \cdot 48 = 96$  configurations within about one month on three PCs. Working on a powerful cluster, of course, reduces this span of time.

### 2.3.8 Number of Wilson eigenvectors

As explained in section 1.2.2 one has to determine  $N$  eigenvectors of the Wilson Dirac operator to speed up the calculation of the overlap operator. Exactly speaking the eigenvectors are needed for the polynomial approximation of the *sgn*-function (12). According to (13) the condition number  $\kappa_c$  for the polynomial approximation depends on the squared ratio of the biggest eigenvalue of the Wilson operator to the  $N + 1$ th Wilson eigenvalue. So for larger  $N$  the condition number  $\kappa_c$  becomes better, and therefore the computation time for the verlap operator is expected to be shorter.

After trying several values of  $N$  the best number of Wilson eigenvectors for the calculation of 200 overlap eigenvalues seemed to be  $N \approx 200$ . Although larger  $N$  means a better  $\kappa_c$ , the program also has to swap more Wilson eigenvectors to the hard disc, so this finally does not result in a shorter computation time.

## 3 Lowest eigenvalues of the overlap operator on a $12^4$ lattice

In the following you will find a presentation of the results of my main task, the numerical calculation of the lowest eigenvalues and eigenmodes of the overlap operator on the  $12^4$  lattice using the overlap program package, which was the largest possible lattice with the available computer power.

The lowest eigenvalues determine the long-range properties of the overlap operator for distances  $> 0.5fm$  (see [IA]). If one replaces the full operator by the truncated operator generated by the lowest eigenvectors, one obtains some kind of an infrared filter, which suppresses ultraviolet noise.

For my calculations I determined the eigenvalues and eigenvectors for 48 gauge configurations at each value of  $\beta = 0.99$  in the confined phase and  $\beta = 1.03$  in the Coulomb phase. In each case three U(1) gauge configurations were combined to one  $3 \times 3$  configuration as described in section 2.3.1, and the 200 smallest eigenvalues for this configuration were obtained numerically using the overlap package written by Thomas Streuer.

As shown in [WE] p. 41 ff. the eigenvalues of the Dirac operator in the Euclidian continuum limit (in the following called  $\lambda$ ) are purely imaginary, whereas the discretization on the lattice destroys this property. So the eigenvalues of the overlap operator come in complex conjugated pairs (in the following  $\lambda_{lat}$  and  $\lambda_{lat}^*$  respectively) and lie on a circle in the complex plane, which was already discussed in subsection 2.3.6.

To obtain  $\lambda$  we have to project  $\lambda_{lat}$  stereographically to the imaginary axis via Möbius transformation (see [TA]):

$$\lambda = \frac{\lambda_{lat}}{1 - \lambda_{lat}/2} \quad (33)$$

In this chapter all eigenvalues are the continuum values  $\lambda$  calculated by this transformation.

### 3.1 Zero-modes

The first point, we want to analyse, is the number of zero-modes in the spectrum of the overlap operator. From the zero-modes of a gauge configuration one can determine its topological charge. Each zero-mode  $\Psi_0$  has a definite positive or negative chirality ([IA]):

$$\gamma_5 \Psi_0 = +\Psi_0 \text{ or } \gamma_5 \Psi_0 = -\Psi_0 \quad (34)$$

(The non-zero modes may have different chirality at different lattice sites  $x$ .)

If we call  $n_+$  the number of zero-modes of a gauge configuration with positive chirality and  $n_-$  with negative chirality, the topological charge is

$$Q = n_- - n_+ \quad (35)$$

As mentioned in [IA] in practical calculations the zero-modes for each gauge configuration all had either positive or negative chirality. In the following we do not distinguish between both cases. Figure 7 confirms the approximately Gaussian shape of the distribution of the absolute value of the topological charge  $|Q|$ . Note that you have to divide the number of configurations with  $|Q| > 0$  by two in comparison to the number of configurations with  $|Q| = 0$ .

In my project I have created 48 gauge configurations at  $\beta = 0.99$  and  $\beta = 1.03$  and calculated the lowest eigenvalues. In table 1 you can see the number of configurations which have 0, 1, 2 or 3 zero-modes. If you want to consider the chirality of each eigenmode, you have to use equation (34) and act  $\gamma_5$  on each eigenvector.

number of zero-modes	$\beta = 0.99$	$\beta = 1.03$
0	17	48
1	18	0
2	12	0
3	1	0

Table 1: Number of configurations with definite number of zero-modes, taken out of 48 configurations for  $\beta = 0.99$  and  $\beta = 1.03$

Obviously there is a difference between the confined phase ( $\beta < \beta_c$ ), where there are several configurations with zero-modes, and the Coulomb phase ( $\beta > \beta_c$ ), where there are absolutely no zero-modes.

The approximately Gaussian shape of the distribution of the topological charge, which I have mentioned above, can be seen in figure 7.

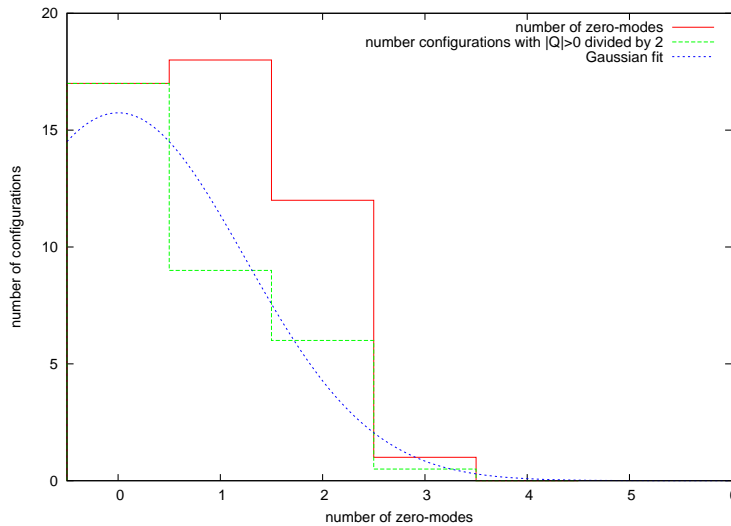


Figure 7: Gaussian distribution of the number of zero-modes for each individual configuration

### 3.2 IPRs

The inverse participation ratio (IPR) is defined as

$$I(\lambda) = V \sum_x \rho_{IPR}(x)^2 \quad (36)$$

$$\text{with } \rho_{IPR}(x) \equiv \sum_a |\Psi_\lambda(x)|^2 \quad (37)$$

where  $V = 12^4$  is the system volume and  $\sum_a$  is the sum over all spinor components of the eigenvector  $\Psi_\lambda(x)$  (see [TA], [IA]).

The IPR is a measure for the spatial distribution of an eigenmode.  $I(\lambda) = 1$  means that  $\Psi_\lambda(x)$  spreads maximally over the entire system, whereas for  $I(\lambda) = V$  the eigenvector  $\Psi_\lambda(x)$  is concentrated on one single site  $x$ .

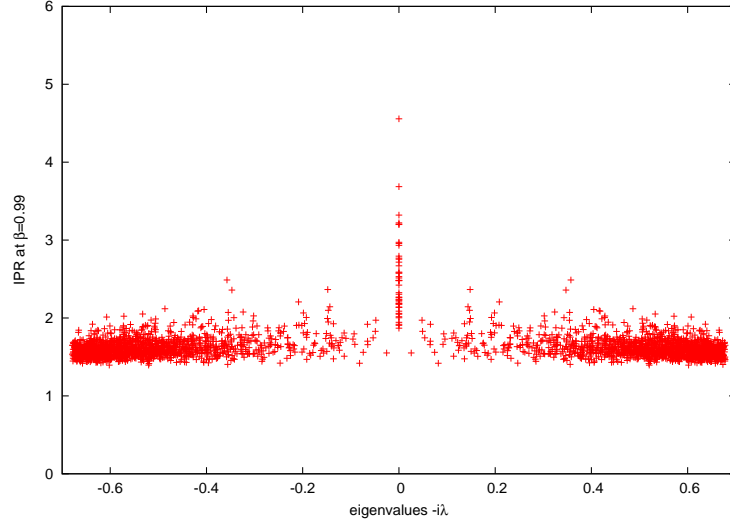


Figure 8: IPR for  $\beta = 0.99$  (48 gauge configurations)

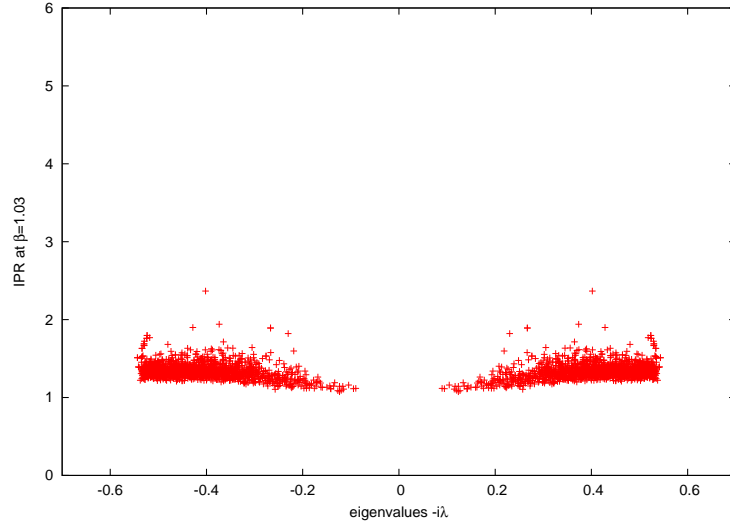


Figure 9: IPR for  $\beta = 1.03$  (48 gauge configurations)

In figures 8 and 9 you can see again, that there are zero-modes for  $\beta = 0.99$  but not for  $\beta = 1.03$ , i.e. above the phase transition in the Coulomb phase. Additionally figure

8 tells us, that zero-modes are more localized than non-zero-modes, because they have a higher IPR.

The average IPR  $\bar{I}$  (average taken over all eigenvalues  $\lambda$ ) for  $\beta = 1.03$  is lower than for  $\beta = 0.99$  as is displayed in figure 10. This means, that for higher  $\beta$  the single eigenmodes  $\Psi_\lambda(x)$  are more uniformly spread over the lattice.

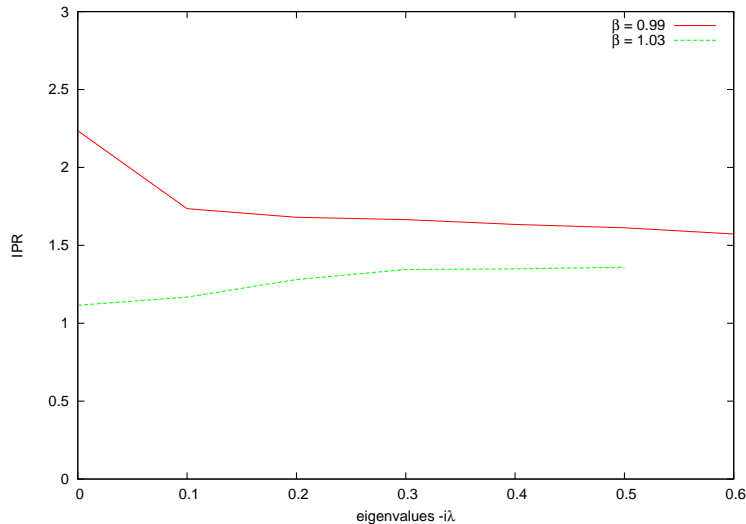


Figure 10: Average IPR for  $\beta = 0.99$  and  $\beta = 1.03$  (average taken over 48 gauge configurations for each value of  $\beta$ )

You can also see, that the density of the smallest eigenvalues becomes higher with increasing  $\beta$ : for  $\beta = 0.99$  we get  $|\lambda_{max}| \approx 0.68$  whereas for  $\beta = 1.03$  we only have  $|\lambda_{max}| \approx 0.55$ . So for higher  $\beta$  the eigenvalues  $\lambda$  are less spread in the IPR diagram, i.e. the density increases with  $\beta$ . (Be careful:  $\lambda_{max}$  is not the biggest eigenvalue, but the biggest one of the 200 lowest eigenvalues.)

### 3.3 Histograms

The IPRs in the former subsection gave us a feeling for the degree of localization of an eigenmode. Now we want to analyse the particular form of such an eigenmode. For this reason we regard histograms  $H_\Psi$  of the eigenfunction-density  $\rho_\Psi(x) \equiv \rho_{IPR}(x)$ , which is defined in (37) (cf. [TA]).

If the eigenfunction  $\Psi$  maximally spreads over the entire system,  $H_\Psi$  will have one sharp peak at  $\rho_\Psi = 1/V \approx 0.00005$ . In contrast a highly localized eigenfunction will have two peaks: a strong one at  $\rho_\Psi \approx 0$  and a very small one at large  $\rho_\Psi$ .

Let us now take a look at the three following histograms in figures 11, 12 and 13. Each of these three histograms is taken as average over 5 eigenmodes.

From the comparison of figure 11 with figure 12, we can see, as in the section before, that zero-modes are more localized than near-zero-modes, because the main peak in

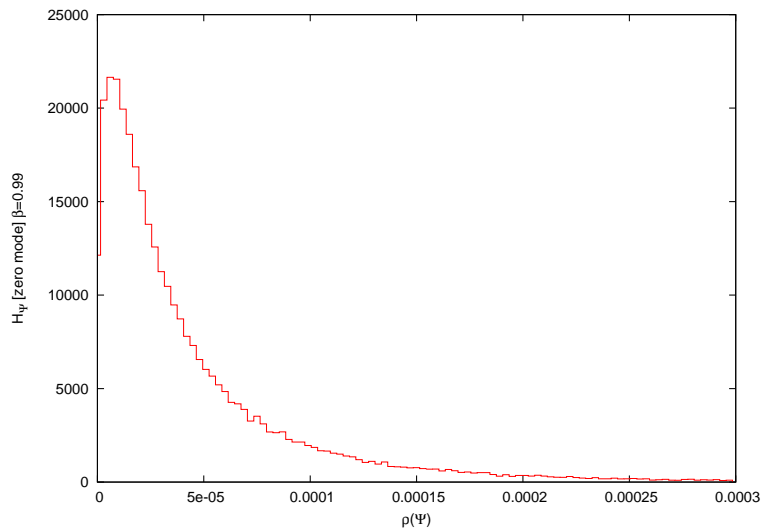


Figure 11: Density of  $\rho_\Psi$  for zero-modes at  $\beta = 0.99$

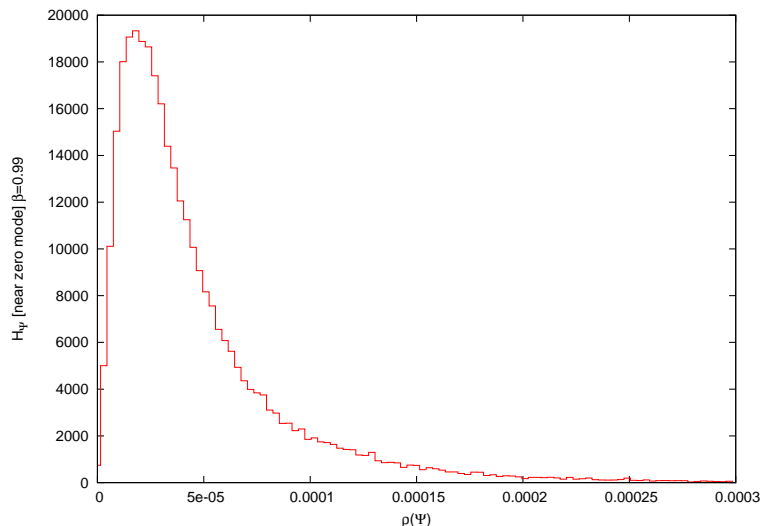


Figure 12: Density of  $\rho_\Psi$  for near-zero-modes at  $\beta = 0.99$

figure 11 is at lower values of  $\rho_\Psi$  than in figure 12. Furthermore the peak for near-zero-modes is wider than the one for exact-zero-modes, which gives also a hint to the higher localization of zero-modes.

In both figures the peak is not at  $\rho_\Psi \approx 5 \cdot 10^{-5}$  but at  $7 \cdot 10^{-6}$  (figure 11) and  $18 \cdot 10^{-6}$  (12) respectively. This means that the corresponding eigenmodes are neither perfectly localized nor maximally delocalized, but something in between.

For a higher value of  $\beta = 1.03$ , i.e. in the Coulomb phase, the eigenmodes are more delocalized, as we have already seen in the former section about IPRs. This can be seen

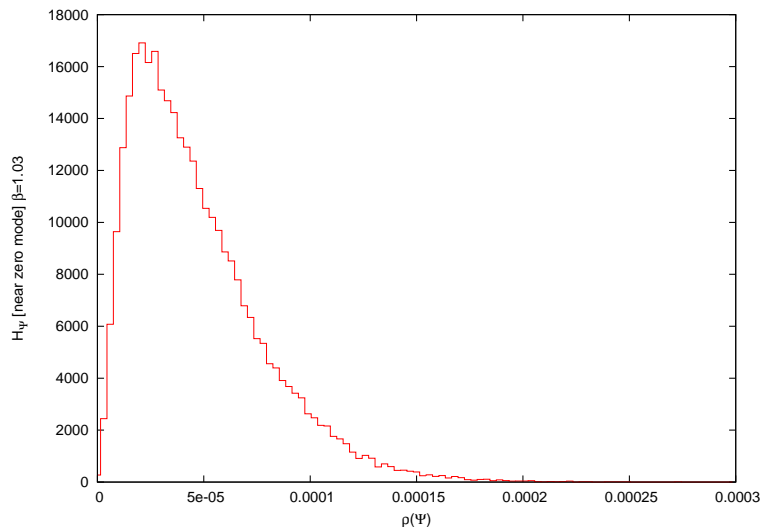


Figure 13: Density of  $\rho_\Psi$  for near-zero-modes at  $\beta = 1.03$

in the histogram in figure 13, where the peak is now wider and shifted in the direction of the value for a uniformly spread eigenmode at  $\rho_{\Psi,unif} \approx 0.00005$ . The tail on the right side of the histogram is also vanishing faster than in the other two diagrams, which means that there are almost no highly-localized contributions to  $\rho_\Psi$ .

### 3.4 Eigenmodes

In figures 14, 15 and 16 you can see slices of the local density  $\rho_\Psi(x)$  of several eigenmodes at different values of  $\beta$ . For each diagram the fixed values of  $z$  and  $t$  are chosen in such a way, that the point  $x_{max}$ , at which the density of the eigenmode achieves its global maximum  $\rho_\Psi(x_{max}) \geq \rho_\Psi(x) \forall x$ , lies within the presented slice of space-time.

In the former sections (3.2 and 3.3) we have learned that eigenmodes for  $\beta = 0.99$  tend to be more localized than eigenmodes in the Coulomb phase at  $\beta = 1.03$ . You will also realize that fact, when you compare figures 14 and 15 with figure 16. The eigenmode in the latter one is more uniformly spread over the volume.

The peak in figure 14 has almost the same height as the peak in figure 15. This is not surprising, as the tails on the right sides of the histograms 11 and 12 also seem to have the same extension. The tail in 13 goes much faster to zero, which is reflected in figure 16, as there is no prominent peak.

## 4 Summary

The basic challenge of this project was the calculation of the eigenvalues of the overlap Dirac operator on a  $12^4$  lattice. Because you have to deal with huge matrices, the numerical treatment of the eigenvalue problem is very ambitious. I performed the calculations using an existing program package for the determination of QCD overlap eigenvalues.



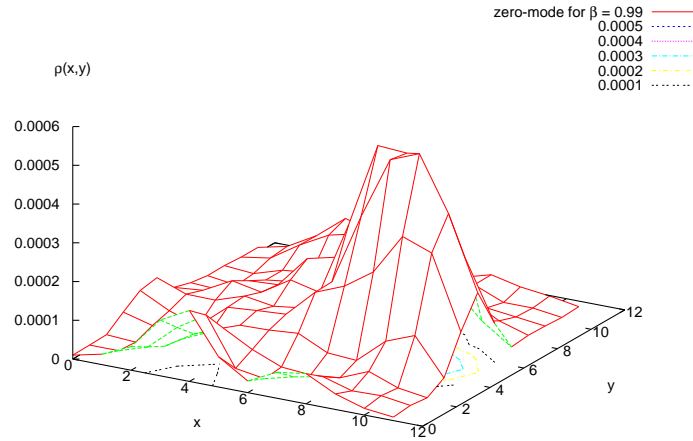


Figure 14: Slice of the local density  $\rho_{\Psi}(x, y)$  of a zero-mode at  $\beta = 0.99$  for constant  $z$  and  $t$

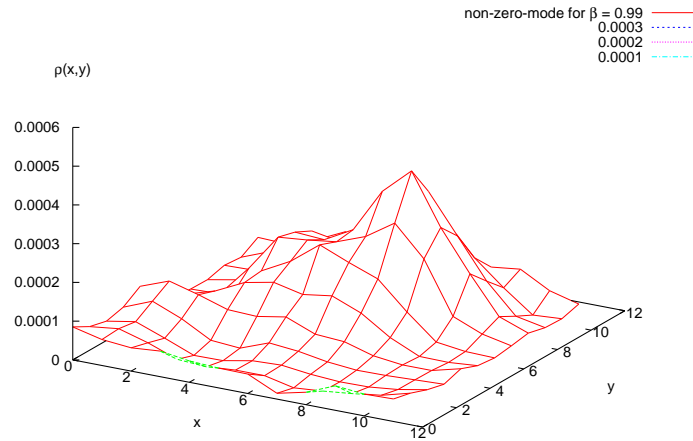


Figure 15: Slice of the local density  $\rho_{\Psi}(x, y)$  of a non-zero-mode at  $\beta = 0.99$  for constant  $z$  and  $t$

In order to do this I had to embed the QED  $U(1)$  gauge field into the  $3 \times 3$  matrices of the QCD algorithm. As I wanted to exclude any errors caused by this embedding, I performed several checks to verify the resulting data. All tests confirmed that the embedding is justified and does not cause any interactions between the individual gauge configurations. So I could use the program for my purpose.

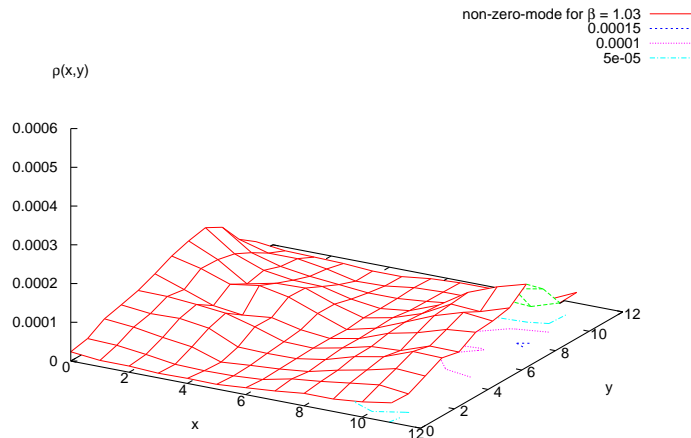


Figure 16: Slice of the local density  $\rho_{\Psi}(x, y)$  of a non-zero-mode at  $\beta = 1.03$  for constant  $z$  and  $t$

After tuning the computation costs to an acceptable span of time by fixing appropriate input parameters I could perform the calculations of the lowest eigenvalues of the overlap operator. In the former chapter you have seen the results, namely the number of configurations with zero-modes, which correspond to the topological charge, and some diagrams displaying the localization of particular eigenmodes.

Last but not least I want to thank Prof. Schäfer and Dr. Gürtler for their patience and helpful explanations. Without their support I would not have mastered the problems in the context of this project. But with their guidance I have learned a lot about lattice calculations and the problems in practical programming, e.g. the need of optimization due to limited computer power.

## 5 Appendix: Input parameters

In tables 2 and 3 you can find the input parameters that I used for the programs of the overlap package. For additional information see also [ST].

parameter	value	remark
version	3	
config	converted.inf	gauge configuration inf-file (required format: 5)
rho	1.6	negative mass of the operator $\rho$
nev	200	number of eigenvectors to be computed
tol	$1.0E - 14$	requested tolerance for the smallest eigenvalues
tol	$1.0E - 14$	requested tolerance for the largest eigenvalues
l_min	0.026	must be larger than the nevth eigenvalue
l_max	2.6	upper bound for the largest eigenvalue
deg	150	degree of polynomial approximation
deg_l	0	not used
lev	0	calculate the largest eigenvalue
lev_g	2.47	tight lower bound for the largest eigenvalue
bc	p	periodic boundary conditions
swap	0	no swapping (available memory was 2 GB)
bs	0	ignored
data_fn	condition	output file
evect_fn	condition_ev	output data-file

Table 2: Input parameters for "condition" (calculation of the eigenmodes of the Wilson Dirac operator)

parameter	value	remark
operator	overlap	requested operator
gaugeconf	converted.inf	gauge configuration inf-file (required format: 5)
evectfile	arnoldi_ov_ev	output file
ovfile	condition	input file with Wilson eigenvalues and eigenvectors (output of "condition")
ov_type	m	Minmax
nev	200	number of eigenvectors to be computed
ncv	400	size of the arnoldi factorization
swap_level	2	necessary swap-level for 2 GB memory
kappa	0.208333333333	$\kappa$
tol	$1.0E - 10$	requested tolerance
ov_error	$1.0E - 9$	error in the overlap approximation
ov_mass	0.0	overlap mass
boundary	p	periodic boundary conditions
which	SM	calculate smallest eigenvalues (capital letters!)

Table 3: Input parameters for "arnoldi" (calculation of the eigenmodes of the overlap operator)

## References

- [TA] Toru T. Takahashi: "Low-lying Dirac eigenmodes and monopoles in 4D compact QED" (2007) [arXiv:hep-lat/0703023v2]
- [KO] Sonja Koller: "Simulation von U(1)-Gittereichtheorie", Universität Regensburg
- [ST] [http://www-zeuthen.desy.de/~streuer/ov\\_doc/](http://www-zeuthen.desy.de/~streuer/ov_doc/) (Thomas Streuer's documentation of the overlap package)
- [ARP] <http://www.caam.rice.edu/software/ARPACK/> (Website of the arnoldi package "ARPACK")
- [WI] [http://en.wikipedia.org/wiki/Arnoldi\\_iteration](http://en.wikipedia.org/wiki/Arnoldi_iteration) (Wikipedia on "Arnoldi iteration")
- [NEF] H. Neff et al.: "On the low fermionic eigenmode dominance in QCD on the lattice" (2001) [arXiv:hep-lat/0106016v2]
- [KR] Daniel Kreßner: "Numerical Methods and Software for General and Structured Eigenvalue Problems" (2004) Berlin
- [GU] Rajan Gupta: "Introduction to lattice QCD" (1998) [arXiv:hep-lat/9807028v1]
- [WE] Volker Weinberg: "Spektrale Eigenschaften der Quantenchromodynamik" (2003)
- [NE1] Herbert Neuberger: "Exactly massless quarks on the lattice" (1997) [arXiv:hep-lat/9707022]
- [NE2] Herbert Neuberger: "More about exactly massless quarks on the lattice" (1998) [arXiv:hep-lat/9801031]
- [NE3] Herbert Neuberger: "The Overlap Dirac Operator" (1999) [arXiv:hep-lat/9910040v1]
- [LU] Martin Lüscher: "Exact chiral symmetry on the lattice and the Ginsparg-Wilson relation" (1998) [arXiv:hep-lat/9802011v1]
- [HA] Peter Hasenfratz et al.: "The index theorem in QCD with a finite cut-off" (1998) [arXiv:hep-lat/9801021v2]
- [BE] Bernd Berg: "Zero-modes of the QED Neuberger Dirac operator", Nuclear Physics B 106 (2002) 592-597
- [IA] Ilgenfritz et al.: "Exploring the structure of the quenched QCD vacuum with overlap fermions" (2007) [arXiv:0705.0018v1 [hep-lat]]