Non-Hermitian Polynomial Hybrid Monte Carlo

DISSERTATION

zur Erlangung des akademischen Grades
doctor rerum naturalium
(Dr. rer. nat.)
im Fach Physik

eingereicht an der
Mathematisch-Naturwissenschaftlichen Fakultät I
der Humboldt-Universität zu Berlin

von
Herrn Dipl.-Phys. Oliver Witzel
geboren am 10.03.1978 in Berlin

Präsident der Humboldt-Universität zu Berlin
Prof. Dr. Dr. h.c. Christoph Markschies

Dekan der Mathematisch-Naturwissenschaftlichen Fakultät I
Prof. Dr. Lutz-Helmut Schön

Gutachter:

1. Prof. Dr. Ulrich Wolff
2. Prof. Dr. Francesco Knechtli
3. Prof. Dr. Anthony D. Kennedy

eingereicht am:	07. Juli 2008
Tag der mündlichen Prüfung:	26. August 2008

Abstract

In this thesis algorithmic improvements and variants for two-flavor lattice QCD simulations with dynamical fermions are studied using the $O(a)$ improved Dirac-Wilson operator in the Schrödinger functional setup and employing a hybrid Monte Carlo-type (HMC) update. Both, the Hermitian and the Non-Hermitian operator are considered.

For the Hermitian Dirac-Wilson operator we investigate the advantages of symmetric over asymmetric even-odd preconditioning, how to gain from multiple time scale integration as well as how the smallest eigenvalues affect the stability of the HMC algorithm.

In case of the non-Hermitian operator we first derive (semi-)analytical bounds on the spectrum before demonstrating a method to obtain information on the spectral boundary by estimating complex eigenvalues with the Lanzcos algorithm. These spectral boundaries allow to visualize the advantage of symmetric even-odd preconditioning or the effect of the Sheikholeslami-Wohlert term on the spectrum of the non-Hermitian Dirac-Wilson operator. Taking advantage of the information of the spectral boundary we design best-suited, complex, scaled and translated Chebyshev polynomials to approximate the inverse Dirac-Wilson operator.

Based on these polynomials we derive a new HMC variant, named non-Hermitian polynomial Hybrid Monte Carlo (NPHMC), which allows to deviate from importance sampling by compensation with a reweighting factor. Furthermore an extension employing the Hasenbusch-trick is derived. First performance figures showing the dependence on the input parameters as well as a comparison to our standard HMC are given. Comparing both algorithms with one pseudo-fermion, we find the new NPHMC to be slightly superior, whereas a clear statement for the two pseudo-fermion variants is yet not possible.

Keywords:
Lattice QCD, Dirac-Wilson Operator, complex Chebyshev Polynomials, Schrödinger Functional, Hybrid Monte Carlo

Zusammenfassung

In dieser Dissertation werden algorithmische Verbesserungen und Varianten für Simulationen der zwei-Flavor Gitter QCD mit dynamischen Fermionen studiert. Dabei wird der $O(a)$-verbesserte Dirac-Wilson-Operator im Schrödinger Funktional sowie ein Update des Hybrid Monte Carlo (HMC)-Typs verwendet. Es wird sowohl der Hermitische als auch der nicht-Hermitische Operator betrachtet.

Für den Hermitischen Dirac-Wilson-Operator untersuchen wir die Vorteile des symmetrischen gegenüber dem asymmetrischen Gerade-Ungerade-Präkonditionierens, wie man durch einen Integrator mit verschiedenen Zeitskalen profitieren kann, sowie welche Auswirkungen die kleinsten Eigenwerte auf die Stabilität des HMC Algorithmus haben.

Im Fall des nicht-Hermitischen Operators leiten wir zunächst eine (semi)-analytische Schranke für das Spektrum her. Anschließend demonstrieren wir eine Methode, um Informationen über den spektralen Rand zu gewinnen, indem wir komplexe Eigenwerte mit dem Lanczos-Algorithmus abschätzen. Diese spektralen Ränder erlauben es, die Vorzüge des symmetrischen Gerade-Ungerade-Präkonditionierens oder den Effekt des Sheikholeslami-Wohlert-Terms für das Spektrum des nicht-Hermitischen Operators zu veranschaulichen. Unter Verwendung der Informationen des spektralen Randes konstruieren wir angepasste, komplexe, skalierte und verschobene Tschebyschow Polynome, um den inversen Dirac-Wilson-Operator zu approximieren.

Basierend auf diesen Polynomen entwickeln wir eine neue HMC-Variante, genannt nicht-Hermitischer polynomialer Hybrid Monte Carlo (NPHMC), welche es erlaubt, vom Importance Sampling unter Kompensation mit einem Gewichtungsfaktor abzuweichen. Desweiteren wird eine Erweiterung durch Anwendung des Hasenbusch-Tricks abgeleitet. Erste Größen der Leistungsfähigkeit, die die Abhängingkeit von den Eingabeparametern als auch einen Vergleich mit unserem Standard-HMC zeigen, werden präsentiert. Im Vergleich der beiden ein-Pseudofermion-Varianten ist der neue NPHMC leicht besser, wohingegen eine eindeutige Aussage im Fall der zwei-Pseudofermion-Variante bisher nicht möglich ist.

Schlagwörter:
Gitter QCD, Dirac-Wilson Operator, komplexe Tschebyschow Polynome, Schrödinger Funktional, Hybrid Monte Carlo

Contents

Chapter 1

Introduction

Our understanding of elementary particles and their interactions is well described by the *standard model* which provides a theoretical formulation for three of the four universal forces in nature. These four forces are *gravitation*, *electromagnetism*, *weak* and *strong* interactions of which gravitation is not included in the standard model. All forces can be described by means of gauge field theories. These theories are based on the gauge principle, i.e. the theory is invariant under local gauge transformations. Moreover, each force is mediated by the corresponding integer spin exchange particle, called *gauge boson*. The gauge bosons are the smallest quantum of that force which can be transferred. As elementary particles we currently know of *leptons* (l) and *quarks* (q), each having three generations with members of spin 1/2 (*fermions*). These are together with the gauge bosons the building blocks of all known matter (see Tab. 1.1), where each particle has an antiparticle, denoted by a bar, with the same mass but opposite charges.

	I	II	III		
quarks	u	c	t	γ	force carriers
	d	s	b	g	
leptons	e^-	μ^-	τ^-	$W^\pm$	
	ν_e	ν_μ	ν_τ	Z^0	

Table 1.1. The constituents of the standard model: three generations of leptons and quarks experiencing forces mediated by the gauge bosons.

Gravitational forces, supposedly carried by the *graviton*, arise by the mass of an object, act attractively but are negligible on the scale of elementary particles. Electromagnetic interactions are transmitted by the massless *pho-*

ton (γ) and affect all (electrically) charged particles like the electron (e^-) or the composite particle proton (p^+). The dynamics of the electromagnetic interactions are formulated by *quantum electrodynamics* (QED), an intensively tested theory showing almost perfect agreement with experiment. Neutral particles like e.g. electron neutrinos (ν_e) do not interact with a photon, but exhibit, as all other particles, interactions with the massive W^+, W^- or Z^0, the three gauge bosons of weak interactions. Whereas for the three mentioned forces we find interactions with leptons and quarks, the strong force only interacts by the exchange of a *gluon* (g) with quarks. Talking collectively of quarks, u, d, c, s, t and b are identified as (quark) flavors. Flavors generalize the well-known concept of isospin, which explains approximate symmetries of composite particles like the proton or the neutron by the almost mass degeneracy of the u- and d-quark. As its name indicates the strong force is on hadronic distances much stronger than the other forces and constrains quarks to be bound to either a quark-antiquark state ($q\bar{q}$) named *meson* or a three quark state (qqq) called *baryon*, jointly denoted as *hadrons*. Two u- and one d-quark form e.g. the familiar proton. Additionally to the electric charge (2/3 for u, c, t and -1/3 for d, s, b) quarks carry the quantum number *color* also denoted as color charge, which one assigns commonly one of the values *red*, *green* and *blue* (r, g, b). Since the strong interactions are so strong they allow to be considered isolated from the other forces when considering hadronic processes and give rise the theory of *quantum chromodynamics* (QCD).

Introductions to quantum field theory and the standard model can be found e.g. in [1] and [2] as well as in many other textbooks on that subject.

1.1 Quantum Chromodynamics

The interactions of quarks and gluons exhibit two remarkable properties: *asymptotic freedom* and *confinement* distinguishing non-Abelian gauge theories, like in particular QCD, from other field theories like QED. Asymptotic freedom refers to the fact that the effective coupling constant of quarks and gluons becomes small at very short distances (large energies). Quarks and gluons can then be considered as quasi free particles being accessible by perturbation theory. Otherwise confinement states that no isolated quarks or gluons can exist leading to the constraint on the quantum number color that only *color singlet* states are allowed. Due to asymptotic freedom we are led to a non-Abelian gauge theory (also known as Yang-Mills theory) with uniquely (by experimental input) determined gauge group $SU(3)$. Adding the constraint of color singlets this theory is called quantum chromodynam-

ics. An analytic derivation of confinement from first principles is not known. The gluon as gauge quanta of QCD carries color charge and is thus able to couple to itself and other gluons. The Lagrangian density of QCD has therefore a gluonic and a fermionic part and is in Minkowski space given by

$$\mathcal{L}_{\mathrm{QCD}} = -\frac{1}{4} F_{\mu\nu} F^{\mu\nu} + \sum_{k=1}^{N_f} \bar{q}_k \left(i \not{D} - m_k \right) q_k. \tag{1.1}$$

The field strength $F_{\mu\nu}(x) = \sum_{a=1}^{8} \lambda^a F^a_{\mu\nu}(x)$ is given by the antisymmetric tensor

$$F^a_{\mu\nu} = \partial_\mu A^a_\nu - \partial_\nu A^a_\mu + g_0 f^{abc} A^b_\mu A^c_\nu, \tag{1.2}$$

with gauge potential $A_\mu(x) = \sum_{a=1}^{8} \lambda^a A^a_\mu(x)$, an Lie-algebra valued anti-Hermitian $SU(3)$ gauge field. Further, g_0 is the strong coupling constant, $\not{D} = \gamma^\mu \left(\partial_\mu + g_0 A_\mu \right)$ the gauge covariant derivative, m_k the bare quark masses, f_{abc} the structure constants and λ^a the generators of the $SU(3)$ color gauge group. The index k runs over all quark flavors, while a is the color index.

Due to the non-perturbative phenomenon of confinement, pure perturbative calculations are only in the limit of high energies successful. In the regime of large energy transfer and weak coupling constant, computations are often performed using the *operator product expansion* (OPE). Factorizing the operator into a "hard" and a "soft" part, the first can be computed perturbatively, whereas the second accounts for non-perturbative effects being parameterized by effective couplings. (For further aspects of perturbative QCD see e.g. [3]) To compute non-perturbative effects from first principles or to access the regime of low energy QCD, a truly *non-perturbative* method is required. Such a method is presented by Wilson and opened the field of *lattice QCD*.

1.2 Lattice Quantum Chromodynamics

Setting up a gauge theory on a four dimensional Euclidean lattice, Wilson shows that in the strong coupling limit confinement arises.[4] As lattice we depict a hypercube of lattice spacing a with periodic boundary conditions (PBC) and get from the standard Minkowski space by means of the Wick rotation to Euclidean space with an imaginary time coordinate. The inverse lattice spacing a serves in addition as *regulator* providing an ultraviolet cutoff.

Transferring the gauge field theory to a lattice formulation it is advantageous to maintain *local gauge invariance*. By coupling matter fields to

the gauge potential A_μ, we achieve local gauge invariance since the gauge transformation between infinitesimally small separated space-time points is transferred (*parallel transporters*). Assigning the quark fields ψ to the lattice sites x we require to gauge connect them by parallel transporters of finite distance. Therefore it appears naturally to bind the gauge degrees of freedom to the links connecting the lattice sites. These link variables, called *gauge field* $U_\mu(x)$, are elements of $SU(3)$ pointing into the four space-time directions μ. A gauge transformation is given by a $SU(3)$ matrix $\Lambda(x)$ rotating color space on each site x. The transformation for $U_\mu(x)$, $\psi(x)$ and $\bar{\psi}(x)$ are

$$\begin{aligned} U_\mu(x) &\to \Lambda(x) U_\mu(x) \Lambda(x+\hat{\mu})^\dagger \\ \psi(x) &\to \Lambda(x)\psi(x) \\ \bar{\psi}(x) &\to \bar{\psi}(x)\Lambda(x)^\dagger. \end{aligned} \tag{1.3}$$

Thus a product $\bar{\psi}(x) U_\mu(x) \psi(x+\hat{\mu})$ is invariant under gauge transformations, as well as an ordered product of gauge links, like the plaquette. In fact, summing over all plaquettes leads to the simplest gauge action,[1] *Wilson's plaquette gauge action*

$$S_G(U) = \frac{1}{g_0^2} \sum_P \mathrm{Tr}\, \{1 - U_P\}\,, \tag{1.4}$$

where P runs over all oriented[2] plaquettes defined by the product

$$U_P(x) = U_\mu(x) \cdot U_\nu(x+\hat{\mu}) \cdot U_\mu(x+\hat{\nu})^\dagger \cdot U_\nu(x)^\dagger. \tag{1.5}$$

Considering the limit of zero lattice spacing we recover the standard continuum Yang-Mills action, the gluonic part in (1.1). Since any lattice action of form (1.4) built by closed loops has this limit the gauge action is not uniquely determined. Finally, we obtain the discretized covariant derivatives

$$\begin{aligned} \widetilde{\nabla}_\mu &= \tfrac{1}{2}(\nabla_\mu + \nabla_\mu^*) \\ \nabla_\mu \psi(x) &= \tfrac{1}{a}\left[U_\mu(x)\psi(x+\hat{\mu}) - \psi(x)\right] \\ \nabla_\mu^* \psi(x) &= -\tfrac{1}{a}\left[\psi(x) - U_\mu^\dagger(x-\hat{\mu})\psi(x-\hat{\mu})\right] \end{aligned} \tag{1.6}$$

After setting up a gauge theory on the lattice it remains to find a prescription how to compute expectation values of a gauge-invariant observable

[1] The bare gauge coupling g_0 is for $SU(N)$ gauge groups often expressed by $\beta = 2N/g_0^2$.
[2] When encountering a link against its orientation the adjoint is to be taken.

$\mathcal{O}(U)$. Transferring Feynman's path integral to the Euclidean lattice an expectation value is obtained by

$$\langle \mathcal{O} \rangle = \frac{1}{\mathcal{Z}} \int \mathcal{D}U \; e^{-S(U)} \; \mathcal{O}(U) \tag{1.7}$$

with the partition function

$$\mathcal{Z} = \int \mathcal{D}U \; e^{-S(U)}, \tag{1.8}$$

where the action $S(U)$ is given for the pure gauge theory by S_G, eq. (1.4). $\mathcal{D}U$ is the product of $SU(3)$ Haar measures of all links of the lattice. Assuming a lattice of eight sites per dimension we have $4 \cdot 8^4 = 16384$ link variables $U_\mu(x)$. Obviously numerical integration can be only performed by Monte Carlo methods resulting in a statistical estimation of (1.7). First a (sufficiently long) sequence of gauge-field configurations with probability distribution $P \propto \exp\{-S(U)\}$ has to be generated. Then the integral is estimated by

$$\langle \mathcal{O} \rangle = \frac{1}{N} \sum_{i=1}^{N} \mathcal{O}(U_i) \tag{1.9}$$

with an error of order $O\left(1/\sqrt{N}\right)$. The job of lattice QCD simulations is now to generate configurations respecting the probability distribution of the theory to be simulated. Unfortunately, only in rare cases *global* heatbath algorithms are available. Instead a given (properly equilibrated) configuration is evolved by *local* updates creating that way a *Markov chain*. Such configurations are not statistically independent and show the effect of autocorrelation. When estimating an observable this has to be taken into account, for further details see Appendix C.2. A valid update algorithm is usually required to obey two conditions: *detailed balance* and *ergodicity*. Detailed balance guarantees that our updating procedure drives us to a canonical (equilibrium) ensemble which has a unique fixed point. Ergodicity means that in configuration space every two configurations must be connected with a non-zero probability or with other words, that the update procedure must be able to reach every point in configuration space with a finite number of steps.

Having now successfully computed an observable $\mathcal{O}$ we still cannot compare the value to experimental data or even to simulations with different lattice actions because the lattice works in addition as regulator which has to be removed. The lattice spacing is moreover the only remaining dimension on the lattice, if all quark masses equal zero. Hence we can compute

either dimensionless ratios of dimensionful quantities or dimensionless numbers constructed from dimensionful quantities and an appropriate power of the cutoff, like for a mass m the product $a \cdot m$. The lattice spacing can be determined by fixing e.g. a mass to experimental data and make then further predictions. If that way one computes e.g. a mass ratio for several lattice spacings by varying the bare parameters, these ratios typically obey

$$\frac{am_1(a)}{am_2(a)} = \frac{m_1(0)}{m_2(0)} + O(a) \text{ effects} \tag{1.10}$$

up to logarithmic corrections arising only if considering higher orders in the running coupling. At leading order (1.10) is independent of the cutoff a. If the $O(a)$ effects are sufficiently small thus the ratio is almost constant for different a, the calculation is said to *scale*. Then we can take the *continuum limit* by extrapolating the values obtained for several lattice spacings to $a = 0$. Instead of a mass m a common choice is to use the *hadronic length* $r_0 \approx 0.5\,\text{fm}$ motivated by phenomenological quark potential models and defined by the force $F(r)$ between two static color sources [5]

$$1.65 = r^2 F(r)\Big|_{r=r_0}. \tag{1.11}$$

A disadvantage of this choice is the large systematic error on r_0 and efforts are undertaken to use an experimentally precisely known quantity like the pion or kaon decay constant.[6, 7]

To explore the behavior of the bare couplings when sending $a \to 0$ we can express each dimensionless combination $a \cdot m(a)$ as functions of the bare coupling, $a \cdot m = f(g_0(a))$. When sending $a \to 0$ the coupling must be tuned such that

$$\lim_{a \to 0} \frac{1}{a} f(g_0(a)) \to \text{const}, \tag{1.12}$$

with the constant given by our mass m, i.e. the "physical" quantities specify how the coupling change when varying a. The evolution of $a\partial g_0(a)/\partial a$ of the running coupling can be determined in perturbation theory, valid for $g_0(a) \to 0$. By the *renormalization group equation* (RGE) one finds a cutoff dependence of the bare coupling at 1-loop

$$\beta(g_0) = -a\frac{\partial g_0}{\partial a} = -\frac{\beta_0}{16\pi^2} g_0^3 \tag{1.13}$$

with $\beta_0 = 11 - (2/3)N_f$ for $SU(3)$. The continuum limit is achieved by sending the bare coupling g_0 to 0. Integrating (1.13) leads to

$$\left(\frac{g_0(a)}{4\pi}\right)^2 = \left(\beta_0 |\ln(a^2\Lambda_{\text{lat}}^2)|\right)^{-1} + \ldots \tag{1.14}$$

where Λ_{lat} specifies some renormalization scheme and is of mass dimension. Computing now a ratio of Λ-parameters we can connect to different renormalization schemes, like e.g. $\overline{MS}$ for which $\log(\Lambda_{\text{lat}}/\Lambda_{\overline{MS}}) = -3.926263\ldots$ for $N_c = N_f = 3$.[8] Due to the sign of β_0 we find a theoretical upper bound on the number of allowed flavors (≤ 16) since otherwise asymptotic freedom does not occur.

Neglecting the fermionic contribution in (1.1) we start by simulating the *pure gauge theory* using the Lagrangian (1.4). The Cabibbo-Marinari update [9] provides us with a very efficient tool. By selecting a proper set of $SU(2)$ subgroups we can update $SU(3)$ link matrices by a local heatbath [10, 11]. In case of growing correlation time in the system, the autocorrelation length increases, too, and the update becomes inefficient (*critical slowing down*). Performing in between (microcanonical) overrelaxation updates [12] we can improve on this problem. Nevertheless we are not simulating QCD.

The next step for more physical applications is to define a Dirac operator to perform e.g. measurements on pure gauge configurations. This is still an uncontrolled approximation commonly denoted as *quenched* since the fermionic part is deleted from the update procedure.[3] However, this approximation leads already to surprisingly good agreement with experiment e.g. for the hadron spectrum.[13, 14]

To arrive at physical simulations, the fermionic part of the action (1.1) has to be included when generating the gauge-field configuration. These *dynamical fermion simulations* are numerically much more challenging and algorithmic development is still ongoing. In particular the huge differences in the quark masses (u, d, s are of order MeV, but c, b, t of order GeV) provide unsolved challenges. Therefore, at first only the two lightest (u and d quarks) are included as mass-degenerate doublet. Although in nature we know of six quarks in our daily life only u and d quarks are dominant. Consequently, the computation of the hadron spectrum with *two-flavor QCD* improves the quenched approximation and leads to convincing agreement with experiment.[15] How two-flavor QCD is realized on the lattice is discussed in the following section.

Despite the fact that the s quark is ten times heavier than the d quark it is still considered as *light* quark and can be treated in a similar way as u and d quark leading to *2+1-flavor QCD*. Here one finds almost perfect agreement for the hadron spectrum.[16] The treatment of the *heavy* quarks (c, b, t) requires a completely different ansatz not discussed here at all. Introductory textbooks on lattice QCD are e.g. [8, 17].

[3] This amounts to setting the fermion determinant equal to 1 as will become apparent in the next section.

1.3 Simulating Dynamical Wilson Fermions

The choice of the action for the fermions is again not unique and different variants exist. As is proven by Nielsen and Ninomiya [18] it is not possible to construct a Dirac operator on the lattice exhibiting all of the desired properties:

small p limit:	the Fourier transformed operator behaves like $\gamma_\mu p_\mu$ for small momentum p_μ
no doublers:	doublers appear within the naive discretization and are removed by appropriately modifying the action
locality:	the range of the action is restricted to be of the same order as the spatial cutoff
chirality:	the Dirac operator anti-commutes with γ_5

We restrict ourselves to ($O(a)$-improved) Wilson fermions which obey a sound theory and are subsequently discussed in some detail. Wilson removed the doublers by adding a second derivative term thus in the continuum they get a mass of order a^{-1}. However, this term breaks explicitly chiral symmetry violating the last point of the *Nielsen-Ninomiya no-go theorem*. The action of the Dirac-Wilson operator[4] is given by

$$S_f = a^4 \sum_x \bar{\psi}(x)(D_W + m_0)\psi(x) \tag{1.15}$$

where $\bar{\psi}$ and ψ are Grassmann-variables carrying color and Dirac indices and are defined at each lattice site $x_\mu = an_\mu$, $n_\mu \in \mathbb{N}^4$. The Dirac-Wilson operator D_W with Wilson parameter $r > 0$ reads

$$D_W = \gamma_\mu \widetilde{\nabla}_\mu - ar\tfrac{1}{2}\nabla_\mu \nabla^*_\mu. \tag{1.16}$$

m_0 is referred to as bare mass and usually $r = 1$ is chosen. If we insert the covariant derivatives (1.6) the action may be written as

$$\begin{aligned} S_f = a^4 \sum_x \Big\{ & \frac{1}{2a} \sum_{\mu=0}^{3} \bar{\psi}(x)\gamma_\mu \left[U_\mu(x)\psi(x+\hat{\mu}) - U^\dagger_\mu(x-\mu)\psi(x-\hat{\mu})\right] \\ & -\frac{r}{2a} \sum_{\mu=0}^{3} \bar{\psi}(x) \left[U_\mu(x)\psi(x+\hat{\mu}) + U^\dagger_\mu(x-\mu)\psi(x-\hat{\mu}) - 2\psi(x)\right] \\ & + m_0 \bar{\psi}(x)\psi(x)\Big\}. \end{aligned} \tag{1.17}$$

Next, we rearrange the terms yielding a diagonal and an off-diagonal part

$$\begin{aligned} S_f = a^4 \sum_x \Big\{ & -\frac{1}{2a} \sum_{\mu=0}^{3} \left[\bar{\psi}(x)(r-\gamma_\mu)U_\mu(x)\psi(x+\hat{\mu}) \right. \\ & \left. + \bar{\psi}(x+\hat{\mu})(r+\gamma_\mu)U_\mu(x)^\dagger\psi(x)\right] + \bar{\psi}(x)\left(m_0 + \tfrac{4r}{a}\right)\psi(x)\Big\}. \end{aligned} \tag{1.18}$$

Rescaling the fields $\bar{\psi}$, ψ by a factor $\sqrt{\frac{2\kappa}{a^3}}$ and defining the hopping parameter $\kappa = (2am_0 + 8r)^{-1}$ we finally achieve

$$S_f = \sum_x \Big\{ -\kappa \sum_{\mu=0}^{3} \Big[\bar{\psi}(x)(r-\gamma_\mu)U_\mu(x)\psi(x+\hat{\mu}) + \bar{\psi}(x+\hat{\mu})(r+\gamma_\mu)U_\mu(x)^\dagger\psi(x) \Big] + \bar{\psi}(x)\psi(x) \Big\}. \tag{1.19}$$

To abbreviate the notation, we introduce the hopping operator H_{xy} and switch to matrix notation

$$H_{xy} = \sum_{\mu=0}^{3} \Big[(r-\gamma_\mu)U_\mu(x)\delta_{x+\hat{\mu},y} + (r+\gamma_\mu)U_\mu(x-\hat{\mu})^\dagger\delta_{x-\hat{\mu},y} \Big]. \tag{1.20}$$

(1.20) suffers from discretization errors which are typical lattice artefacts. Due to the fact that lattice actions are not uniquely determined it is possible to reduce these artefacts by *improving* the action still yielding the same continuum limit. Symanzik introduced additional terms proportional to powers of the lattice spacing a in order to compensate lattice artefacts.[19, 20] Since higher order terms become more and more complicated only $O(a)$ improvement is applied to the Dirac-Wilson operator resulting in the additional Sheikholeslami-Wohlert-term [21] of the $O(a)$ improved Wilson's fermion action

$$S_f = \sum_{x,y} \bar{\psi}(x) \left[\left(1 + \frac{i}{2} c_{\mathrm{sw}} \kappa \sigma_{\mu\nu} \mathcal{F}_{\mu\nu}(x) \right) \delta_{xy} - \kappa H_{xy} \right] \psi(y) \tag{1.21}$$

$$= \sum_{x,y} \bar{\psi}(x) M_{xy} \psi(y), \tag{1.22}$$

where M_{xy} is named fermion matrix. The Sheikholeslami-Wohlert-term carries the improvement constant c_{sw} and is built up by the clover shaped contributions of the field-strength tensor

$$\begin{aligned} \mathcal{F}_{\mu\nu}(x) = \frac{1}{8} \Big[\Big(& U_\mu(x)U_\nu(x+\hat{\mu})U_\mu^\dagger(x+\hat{\nu})U_\nu^\dagger(x) \\ & + U_\nu(x)U_\mu^\dagger(x+\hat{\nu}-\hat{\mu})U_\nu^\dagger(x-\hat{\mu})U_\mu(x-\hat{\mu}) \\ & + U_\mu^\dagger(x-\hat{\mu})U_\nu^\dagger(x-\hat{\mu}-\hat{\nu})U_\mu(x-\hat{\mu}-\hat{\nu})U_\nu(x-\hat{\nu}) \\ & + U_\nu^\dagger(x-\hat{\nu})U_\mu(x-\hat{\nu})U_\nu(x-\hat{\nu}+\hat{\mu})U_\mu^\dagger(x) \Big) - \mathrm{H.c.} \Big]. \end{aligned} \tag{1.23}$$

For our considerations it turns out to be advantageous to write the fermion matrix as $M_{xy} = \delta_{xy} - K_{xy}$, with

$$K_{xy} = \kappa \left[H_{xy} - \frac{i}{2} c_{\mathrm{sw}} \sigma_{\mu\nu} \mathcal{F}_{\mu\nu}(x) \delta_{xy} \right]. \tag{1.24}$$

Looking at eq. (1.21) we observe on the one hand that the hopping operator is γ_5-Hermitian, $\gamma_5 H_{xy}\gamma_5 = H_{xy}^\dagger$, and for L and T even holds the similarity transformation $(-1)^{\sum_{\mu=0}^{3} x_\mu} H_{xy} (-1)^{\sum_{\mu=0}^{3} y_\mu} = -H_{xy}$. The clover term obeys on the other hand also the γ_5-Hermiticity but shows no sign-flip for the second transformation. Hence only the γ_5-Hermiticity is present for K_{xy}. Furthermore, we note that $\mathcal{F}_{\mu\nu}$ is antisymmetric and anti-Hermitian $\mathcal{F}_{\mu\nu} = -\mathcal{F}_{\nu\mu} = -\mathcal{F}_{\mu\nu}^\dagger$.

Since the Grassmann variables are not suitable for a simulation on the lattice we continue by integrating out the Grassmann variables which leaves us with the determinant on the fermion matrix

$$\mathcal{Z} = \int \mathcal{D}U \ \mathrm{e}^{-S_G(U)} \ \det\{M\}. \tag{1.25}$$

Incorporating this determinant is the challenge of simulating dynamical fermions. Since a numerical computation of $\det\{M(U)\}$ is impractical we represent the determinant as a bosonic Gaussian integral. This requires $\det\{M\}$ to be positive. Therefore, it is convenient to consider $\det\{M^\dagger(U)M(U)\}$ instead of $\det\{M(U)\}$ with the interpretation of two (degenerate) flavors and make use of the property $\det\{M(U)\} = \det\{M^\dagger(U)\}$.

Now the determinant can be estimated by an integral over bosonic pseudo-fermion fields $\phi^\dagger$, ϕ leading to the partition function

$$\mathcal{Z} = \int \mathcal{D}U \, \mathcal{D}\phi^\dagger \, \mathcal{D}\phi \ \mathrm{e}^{-S_G(U) - S_B(U,\phi^\dagger,\phi)}, \tag{1.26}$$

with the bosonic action for two-flavor Wilson-fermions given by

$$S_B(U,\phi^\dagger,\phi) = \phi^\dagger (MM^\dagger)^{-1} \phi. \tag{1.27}$$

Since the bosonic action arises from the determinant of the fermion matrix we are free to multiply M by γ_5 and yield that way an Hermitian operator.

The introduced periodic boundary conditions are only used to verify some properties of the Dirac-Wilson operator. Our main interest are *Schrödinger functional* (SF) boundary conditions, where the spatial directions are taken to be periodically continued, but the time direction has fixed Dirichlet boundary conditions. Hence instead of a hypercube we have a $L^3 \times T$ space-time cylinder.[22–24] The SF has some advantages over periodic BC in all directions (torus). As we demonstrate in Chapter 4 the SF has an infrared cutoff proportional to T^{-1}. The fermionic boundary fields can serve as source for correlation functions and by the gluonic boundary fields a background field can be induced.

1.4 Recent Challenges

Despite the steady progress in developing new machines providing more and more computational resources, dynamical fermion simulations remain challenging. Given the chance of higher computational power, numerical simulations of lattice QCD become more ambitious by simulating at lower quark masses and in bigger volumes. The simulations are hence becoming more physical. Employing formerly used algorithms (without improvements) at these new parameters proves not to be suitable. One problem of hybrid Monte Carlo-type algorithms (see Section 5) is e.g. the increase of large energy violations (spikes) as shown in Fig. 1.1. These lower the acceptance rate and cause an increase in the autocorrelation time of observables. Further, the correctness of the algorithm becomes doubtful due to reversibility violations.

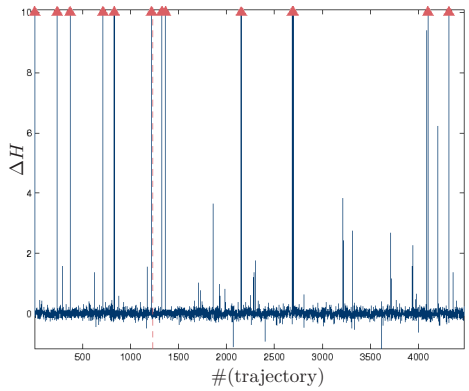

Figure 1.1. Energy violations in a standard HMC simulation (Run C1 cf. Section D.2). The red triangles indicate spikes in ΔH of up to 4383.

In the focus of this thesis are algorithmic improvements for two-flavor lattice QCD simulations employing the Dirac-Wilson operator. On the one hand we consider a standard HMC run with the Hermitian Dirac-Wilson operator, and on the other we develop a new HMC-variant named *non-Hermitian polynomial hybrid Monte Carlo* (NPHMC) which is based on a polynomial approximation of the non-Hermitian operator. This polynomial approximation relies on complex, scaled and translated Chebyshev polynomials. In Chapter 2 we introduce those Chebyshev polynomials, derive the approximation of the non-Hermitian operator and show that the obtained simple recurrence relations are stable. Next we present a common and important tool to enhance the efficiency of dynamical fermion simulation: *even-odd preconditioning*. It can be applied to both, the Hermitian and the non-Hermitian, operator. Due to the Sheikholeslami-Wohlert term there exist two versions, leading to the

asymmetric or *symmetric* even-odd preconditioned Dirac-Wilson operator. Both are given in Chapter 3.

Following these general remarks we study spectral properties of the non-Hermitian Dirac-Wilson operator as it is used in dynamical fermion simulations (Chapter 4). We start simple by recalling first bounds on the spectrum for the free operator with periodic boundary conditions and proceed then to the analytically not exactly solvable case of Schrödinger functional boundary conditions. Next we derive a relation between the preconditioned and unpreconditioned operators before investigating properties of the approximation by Chebyshev polynomials, in particular the rate of convergence. As mentioned above the non-Hermitian Dirac-Wilson operator is non-normal and hence powerful algebraic theorems, like the spectral theorem, are absent. We comment on this and provide in the Appendices A and B mathematical background on norms of matrices and how to compute eigenvalues of non-Hermitian matrices. Computing these eigenvalues we are enabled to estimate the spectral boundary and explain e.g. why symmetric even-odd preconditioning is advantageous or visualize the effect of the $O(a)$ improvement term.

Before employing our new knowledge on how to obtain well working approximations on the inverse Dirac-Wilson operator we review in Chapter 5 the important aspects and ingredients of the HMC algorithm, explain some common modifications and discuss three variants of particular interest. In the following Chapter 6 we extend this list by deriving our new variant NPHMC. Moreover, we show how to incorporate a widely used modification, the Hasenbusch-trick.

After these theoretical remarks we turn to the numerical side. First we return to our standard HMC algorithm (Chapter 7) and demonstrate that the symmetric version of even-odd preconditioning is also in case of the Hermitian operator advantageous and how one can gain from multiple time scale integration. We continue by analyzing the performance of large volume simulations in particular with respect to the autocorrelation times of observables and analyze the distribution of the smallest eigenvalue as tool to determine the stability of the algorithm. This section is closed by a scaling test on cutoff effects of non-perturbatively renormalized quantities. In Chapter 8 we investigate the dependence of our new algorithm on the polynomial input parameters and explain how to set/tune them. In addition first cost and performance figures are computed and finally we try to compare the performance of the new NPHMC with our standard HMC.

Reviewing this work we conclude in Chapter 9 and highlight the most important results. Furthermore, we indicate a future extension to simulate $2 + 1$ dynamical flavors, an option providing another motivation to develop this new HMC-variant.

Chapter 2

Polynomial Approximation

The key ingredient of this new update algorithm is to approximate the inverse, non-Hermitian Dirac-Wilson operator by a polynomial

$$P_n(M) \approx M^{-1} = (\mathbb{1} - K)^{-1} \tag{2.1}$$

to be computed by simple recursions only. Since the spectrum of M is not known we seek approximations on a region of the complex plane covering the spectrum. Guided by the free theory, these regions are described well by a bounding ellipse and we first recall some geometric properties of the ellipse and introduce our notation. Next we discuss the simpler case of an approximation by a geometric series which corresponds to a circular bound. After introducing complex Chebyshev Polynomials we derive an improved approximation. We close this section by verifying the stability of the obtained recurrence relations which later are used to implement the polynomial approximations.

2.1 The Ellipse

As can be seen in Fig. 2.1 an ellipse is characterized by its two half axis a and b. Without loss of generality we assume $a > b$. Then a is named *major half axis*, while b is called *minor half axis*. If one considers the limit of $a = b$ the ellipse becomes a circle of radius $r = a = b$. For vanishing b the ellipse degenerates to a line. Defining the eccentricity

$$e = \sqrt{a^2 - b^2} \tag{2.2}$$

we receive a measure on the deviation of the ellipse form a circle ($e = 0$) or a line ($e = a$). Geometrically the eccentricity is the distance of the focal point

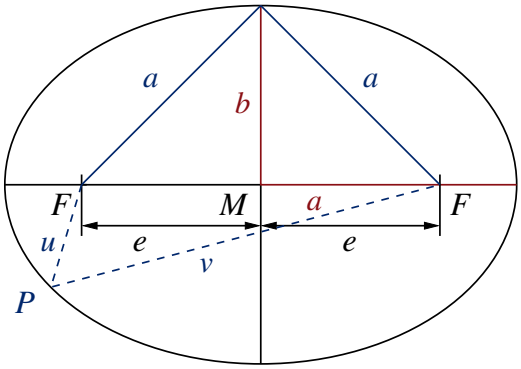

Figure 2.1. Sketch of an Ellipse with its parameters.

F from the center M. For each point P on the ellipse holds the *defining equation of an ellipse*

$$2a = u + v, \tag{2.3}$$

This equation allows further a simple construction of an ellipse by tethering a string of length $2a$ at both ends to the focal points and drawing two half ellipses keeping the string always tightened (*gardeners construction method*). Finally, we can parameterize an ellipse by

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a\cos(t) \\ b\sin(t) \end{pmatrix}, \qquad \text{for} \quad t = 0, \ldots, 2\pi. \tag{2.4}$$

2.2 Geometric Series

If the spectrum of K lies within the unit circle, $\|K\| < 1$, we can employ a truncated geometric series as simple approximation of M^{-1} [25]

$$P_n^{\text{Geo}} = \sum_{j=0}^{n} K^j \tag{2.5}$$

and we find in the limit $n \to \infty$

$$\lim_{n\to\infty} P_n^{\text{Geo}} = (\mathbb{1} - K)^{-1} = M^{-1}. \tag{2.6}$$

By truncating the series at n our approximating polynomial deviates from the true inverse and we define the remainder to get a quantity of the quality of our approximation

$$R_{n+1} = \mathbb{1} - M P_n^{\text{Geo}} = K^{n+1}. \tag{2.7}$$

Obviously, eq. (2.5) and (2.7) allow for a trivial one-step recursion which converges and is hence numerically stable as long as $\|K\| < 1$.

2.3 Chebyshev Polynomials

We can improve the approximation by using complex Chebyshev polynomials. Considering first the two complex variables $\theta = \zeta + i\varphi$ and $z = x + iy$ with the mapping

$$\begin{aligned} z &= \cosh(\theta) \\ &= \cosh(\zeta + i\varphi) = \cosh(\zeta)\cos(\varphi) + i\sinh(\zeta)\sin(\varphi) \end{aligned} \tag{2.8}$$

we can identify

$$x = \cosh(\zeta)\cos(\varphi) \quad \text{and} \quad y = \sinh(\zeta)\sin(\varphi). \tag{2.9}$$

By the well-known identity $\cos^2\varphi + \sin^2\varphi = 1$ we get

$$\frac{x^2}{\cosh(\zeta)^2} + \frac{y^2}{\sinh(\zeta)^2} = 1. \tag{2.10}$$

Hence the line $\zeta = \text{const} > 0$ is mapped onto an ellipse $\mathcal{E}(\zeta)$ with semi-major axis $|\cosh(\zeta)|$ and semi-minor axis $|\sinh(\zeta)|$, while the foci are at -1 and 1. The elliptical disc bounded by $\mathcal{E}(\zeta)$ is denoted as $\mathcal{D}(\zeta)$. For $0 < \alpha < \beta$ the mapping yields for $\zeta = \alpha$ an ellipse inside and confocal to the ellipse created by the mapping at $\zeta = \beta$ (cf. Fig. 2.2).

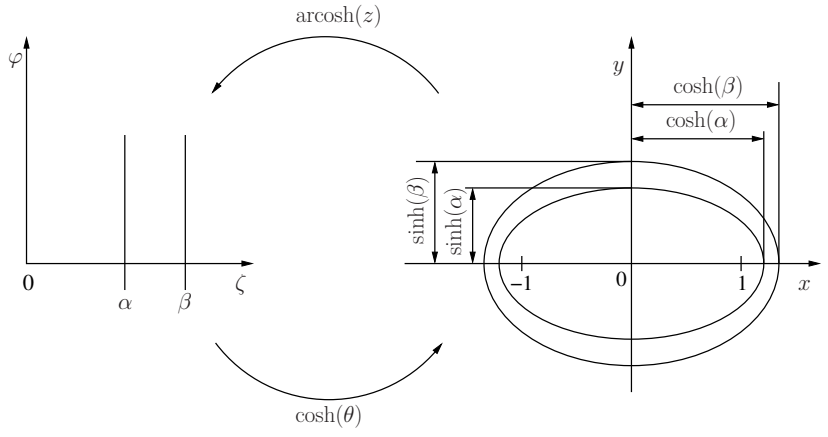

Figure 2.2. Mapping from the θ-plane to the z-plane and back by the cosh and arcosh function, respectively.[26]

The complex Chebyshev polynomials are defined by

$$T_n(z) = \cosh(n \operatorname{arcosh}(z)), \qquad n = 0, 1, 2, \ldots \tag{2.11}$$

Thus the n^{th} Chebyshev polynomial maps an ellipse onto a vertical line segment in $0 \leq \mathrm{Re}\,\{z\};\ 0 \leq \mathrm{Im}\,\{z\} < 2\pi$, multiplies this line segment by n and maps the new line segment back onto another ellipse.[26]

Moreover, we can derive from eq. (2.11) the recursion relation for Chebyshev polynomials

$$T_{n+1}(z) = 2zT_n(z) - T_{n-1}(z), \tag{2.12}$$

supplemented by the two initial values

$$T_0(z) = 1 \quad \text{and} \quad T_1(z) = z.$$

Before seeking a polynomial approximation Q_n of the inverse of a matrix A using (2.11) we obtain some constraints on the polynomials. First, the polynomials must obey a recurrence relation such that not all previous degrees of the polynomial need to be stored and the approximation is of practical use. Secondly, we like to choose polynomials obeying

$$\|e_n\| \leq \|Q_n(A)\|\,\|e_0\|, \tag{2.13}$$

where e_i is the error of the i^{th} step and $Q_n(z)$ is such that $Q_n(0) = 1$. Third, the spectrum of A must show the following properties:
If the matrix A is normal (cf. Appendix A.2), A can be diagonalized by a unitary transform and hence the *Jordan normal form* of A is also diagonal, i.e. every *Jordan block* with eigenvalue λ_i is of size 1×1

$$Q_n(A) = Q_n(S^{-1}JS) = S^{-1}Q_n(J)S \quad \text{with } J = \mathrm{diag}(\lambda_1, \lambda_2, \ldots), \tag{2.14}$$

and for J diagonal we have

$$Q_n(J) = \begin{bmatrix} Q_n(\lambda_1) & & \\ & \ddots & \\ & & Q_n(\lambda_k) \end{bmatrix}. \tag{2.15}$$

Therefrom follows that for $\|Q_n(A)\| \to 0$ as $n \to \infty$ it must hold for *all* λ_i that $P_n(\lambda_i) \to 0$ as $n \to \infty$.

For non-normal A the Jordan blocks are non-trivial thus the Jordan normal form becomes $A = S^{-1}JS$ with

$$J = \begin{bmatrix} J_1 & & \\ & \ddots & \\ & & J_k \end{bmatrix} \quad \text{and} \quad J_i = \begin{bmatrix} \lambda_i & 1 & \\ & \ddots & \ddots \\ & & \lambda_i \ \ 1 \\ & & \lambda_i \end{bmatrix}, \tag{2.16}$$

where J_i is the Jordan block associated with the eigenvalue λ_i and of dimension d_i. $Q_n(J)$ is then given by

$$Q_n(J) = \begin{bmatrix} Q_n(J_1) & & \\ & \ddots & \\ & & Q_n(J_k) \end{bmatrix} \tag{2.17}$$

$$\text{with} \quad Q_n(J_i) = \begin{bmatrix} Q_n(\lambda_i) & Q_n'(\lambda_i) & \frac{1}{2!}Q_n''(\lambda_i) & \dots & \frac{1}{d_i!}Q_n^{d_i-1}(\lambda_i) \\ & \ddots & \ddots & & \vdots \\ & & \ddots & \ddots & \frac{1}{2!}Q_n''(\lambda_i) \\ & & & \ddots & Q_n'(\lambda_i) \\ & & & & Q_n(\lambda_i) \end{bmatrix}$$

Hence for non-normal A we have to ensure that $Q_n(\lambda_i)$ and all derivatives $Q_n^{(j)}(\lambda_i)$ for $j < d_i$ go to zero as $n \to \infty$ such that $\|Q_n(A)\| \to 0$ as $n \to \infty$. This forces the constraint that the polynomials Q_n and their derivatives must be small on the spectrum of A.

Since we do not know the spectrum of A exactly, we choose polynomials that are small on an elliptical region containing the spectrum. These are the scaled and translated Chebyshev polynomials [26]

$$Q_n(\lambda) = \frac{T_n\left(\frac{d-\lambda}{e}\right)}{T_n\left(\frac{d}{e}\right)}. \tag{2.18}$$

These polynomials provide the *optimal* approximation with respect to the L_∞ norm, i.e. they achieve

$$\min_{Q_n} \max_{\lambda \in \mathcal{D}(x)} |Q_n(\lambda)| \tag{2.19}$$

under the constraint that the spectrum is contained in a region not including the origin and $0 < e \leq d$.[26][1]

2.4 Chebyshev Approximation

After introducing in the previous section complex Chebyshev polynomials we derive now an improved approximation of the non-Hermitian Dirac-Wilson operator. Here we have to start from the remainder since this obeys by construction the constraint to be small.[28] Expressing R_{n+1} as scaled and

[1] The statement by Manteuffel that the Chebyshev polynomials provide an *optimal* approximation is at least questionable, see [27].

translated Chebyshev polynomials (2.18) we find

$$R_{n+1}(M) = \mathbb{1} - M P_n^{\mathrm{Cby}} = \frac{T_{n+1}((1-M)/e)}{T_{n+1}(1/e)}. \tag{2.20}$$

Due to the two step recurrence relation of the Chebyshev polynomials (2.12) we can derive starting from eq. (2.20) a recursive description for the R_{n+1} and the P_n, too,

$$R_{n+1}(M) = a_n K R_n(M) + (1-a_n) R_{n-1}(M) \tag{2.21}$$
$$\text{with} \quad R_1(M) = K; \quad \text{and} \quad R_0(M) = \mathbb{1}$$
$$P_n(M) = a_n(\mathbb{1} + K P_{n-1}(M)) + (1-a_n) P_{n-2}(M) \tag{2.22}$$
$$\text{with} \quad P_1(M) = a_1(\mathbb{1} + K) \quad \text{and} \quad P_0(M) = \mathbb{1}\,.$$

Here we introduced the coefficients a_n defined by

$$a_n = \frac{2}{e} \frac{T_n(1/e)}{T_{n+1}(1/e)} \tag{2.23}$$

and obeying the recursion

$$a_n = (1 - e^2 a_{n-1}/4)^{-1} \quad \text{with} \quad a_1 = (1 - e^2/2)^{-1}. \tag{2.24}$$

Considering the limit of $e \to 0$ the ellipse degenerates to a circle, a_n approaches 1 and the Chebyshev approximation falls back to the approximation by a geometric series. Although the defining equation (2.20) does not allow for $e = 0$, both recurrence relations, eq. (2.21) and (2.22) are well defined in that limit and one recovers the equations of the geometric series (2.7) and (2.5), respectively.

2.5 Stability of the Recurrence Relations

Numerical methods face the risk of instabilities due to the fact that floating point numbers have limited precision and round-off errors occur. Also with 64-bit arithmetics these errors may grow exponentially in recurrence relations. Therefore, we have to verify all recursive formulae to be numerically stable since stability is a property of the recurrence relation.[29]

We start by analyzing the one-step recursion for the a_n's, eq. (2.24), where e is small and real[2], hence $0 < e^2/4 \ll 1$. Computing the difference of

[2]The ellipse with max. e has vanishing minor half axis b. Since the spectrum must be within the unit circle (convergence of the geom. series) we know $a < 1$ and it follows $e < 1$.

successive elements the convergence is shown and we can determine the limit for $n \to \infty$

$$A = \lim_{n\to\infty} a_n = \lim_{n\to\infty} \left(1 - \frac{e^2}{4} a_{n-1}\right)^{-1} = \frac{2}{e^2}\left(1 - e\sqrt{e^{-2}-1}\right). \tag{2.25}$$

In the following we discuss the stability of two-step recursions and introduce some technical terms from numerical mathematics.[30, 31] The Chebyshev polynomials obey the *homogeneous* and *symmetric* three-term recurrence relation

$$y_{n+1}(z) - 2z\, y_n(z) + y_{n-1}(z) = 0. \tag{2.26}$$

(2.26) is called homogeneous since the constant term vanishes and it is symmetric under the change of y_{n+1} and y_{n-1}. To obtain the *characteristic equation* of the recursion we impose the ansatz[3]

$$y_n = \lambda^n \tag{2.27}$$

and yield

$$\lambda^{n-1}\left(\lambda^2 - 2z\,\lambda + 1\right) = 0. \tag{2.28}$$

Besides the trivial solution of (2.28) we find two eigenvalues

$$\lambda_\pm = z \pm \sqrt{z^2 - 1}. \tag{2.29}$$

Due to (2.27) we claim that $y_n = \lambda_+^n$ and $y_n = \lambda_-^n$ are both linearly independent solutions of (2.26) as well as the linear combination

$$y_n = c_+\lambda_+^n + c_-\lambda_-^n. \tag{2.30}$$

The coefficients $c_\pm$ are determined by exploiting the initial values of the recurrence, $y_1 = z$ and $y_0 = 1$,

$$c_\pm = \pm\frac{z - \lambda_\mp}{\lambda_+ - \lambda_-} = \frac{1}{2}. \tag{2.31}$$

Eq. (2.30) can be solved using the previously obtained results and setting $z = \cosh\theta$. Then $\lambda_\pm = \mathrm{e}^{\pm\theta}$ and

$$y_n = \cosh(n\theta) = \cosh(n\,\mathrm{arcosh}(z)) = T_n(z). \tag{2.32}$$

[3] A justification is provided using function analysis. But we may tolerate this step and check afterwards the result to be correct.

On the one hand, the last step shows that the Chebyshev polynomials T_n are formed by a combination of the two linear independent solutions of the recurrence relation; on the other hand, since we know that the Chebyshev polynomials obey the recursion the result validates the ansatz (2.27).

Now we are well-equipped to address the question of stability. First, we consider the ratio of the two linearly independent solutions in the limit of n going to infinity. Obviously, this depends on the value of θ or z, respectively,

$$\lim_{n\to\infty} \frac{\lambda_-^n}{\lambda_+^n} = \lim_{n\to\infty} \frac{\mathrm{e}^{-\theta n}}{\mathrm{e}^{+\theta n}}. \tag{2.33}$$

Let us assume θ to be real and positive then λ_+^n grows without limit while λ_-^n becomes smaller and smaller hence the ratio in (2.33) goes to 0 as $n \to \infty$. Therefore, λ_+^n is called *dominant solution* and λ_-^n is named *minimal solution.*[30] Dominant solutions can be calculated using forward recursions, whereas this is numerically impossible for minimal solutions.[4] In case of θ being negative, λ_+ becomes the minimal and λ_-^n the dominant solution. Considering complex values for θ we observe that only the real part determines the stability. The imaginary part contributes like a phase and forces the value to oscillate without effecting the stability.

Returning to our Chebyshev polynomials we compute for $\mathrm{Re}\{\theta\} \geq 0$ the limit

$$\lim_{n\to\infty} \frac{T_n}{\lambda_+^n} = \lim_{n\to\infty} \frac{\lambda_+^n + \lambda_-^n}{2\lambda_+^n} \to \frac{1}{2}, \tag{2.34}$$

and for $\mathrm{Re}\{\theta\} \leq 0$

$$\lim_{n\to\infty} \frac{T_n}{\lambda_-^n} = \lim_{n\to\infty} \frac{\lambda_+^n + \lambda_-^n}{2\lambda_-^n} \to \frac{1}{2} \tag{2.35}$$

and conclude that the recursion is always stable because in any case the dominant solution is part of the linear combination. Looking at (2.33) the "worst" case is observed to be $|\mathrm{Re}\{\theta\}| < 1$. Then the exponential growth (decay) is slower, dominant and minimal solutions are less distinct, and hence both errors may add. Nevertheless the recursion remains stable!

Furthermore, we verify this numerically in `Matlab`. There we demonstrate the exponential growth or decay of the two linearly independent solutions and show in addition the behavior of their linear combination in relation to computing Chebyshev polynomials by, e.g., exponential expressions (cf. Figure

[4] Possibilities to compute minimal solutions are provided e.g. in terms of continued fractions.[30]

2.3). T_n follows the dominant solution λ_+ — regardless that around $n = 20$ the computation of the minimal solution breaks down. Checking the relative error of T_n we find it is on machine precision like the error of the dominant solution λ_+. Only λ_- shows the expected catastrophic deviations.

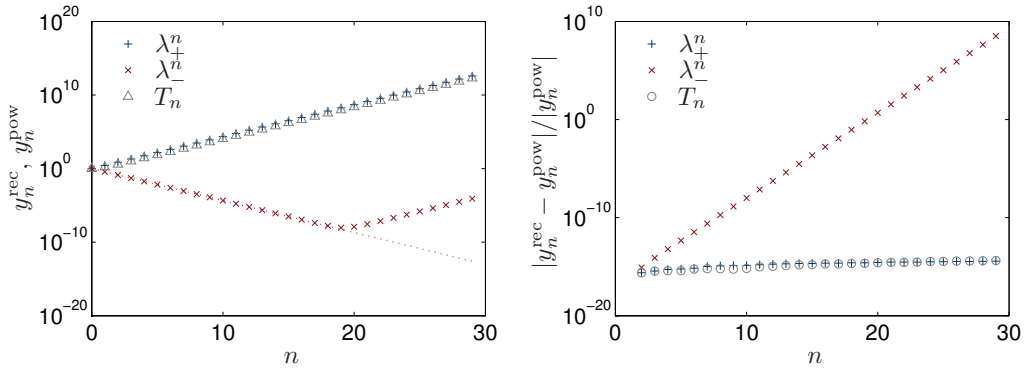

Figure 2.3. Left: numerical result of employing (2.26) to compute λ_+^n, λ_-^n, and $T_n = (\lambda_+^n + \lambda_-^n)/2$ for $\theta = 1.0$. The dotted line indicate the "true" behavior computing $\exp\{-\theta n\}$. Right: the stable recursions show an error close to machine precision whereas λ_-^n becomes dominated by λ_+^n resulting in catastrophic deviations.

We proceed and turn our attention to the recursion of the remainder.[5] Equation (2.21) is not symmetric but still homogeneous. However, we can continue as before and obtain the characteristic equation by setting $R_n = \lambda^n$

$$\lambda^{n-1}\left(\lambda^2 - AK\lambda - (1-A)\right) = 0. \tag{2.36}$$

Determining the eigenvalues we find $\lambda_\pm = AK/2 \pm \sqrt{(1-A) + (AK)^2/4}$ and specify the constants $c_\pm$ of the general solution, $R_n = c_+\lambda_+^n + c_-\lambda_-^n$, by evaluating R_n at the initial values $R_0 = 1$ and $R_1 = K$. Hence the general solution reads

$$R_n = \frac{1}{\lambda_+ - \lambda_-}\left((K-\lambda_-)\lambda_+^n - (K-\lambda_+)\lambda_-^n\right). \tag{2.37}$$

Again we find λ_+^n (λ_-^n) to be the dominant (minimal) solution (or vice versa) with the conclusion that the linear combination giving R_n is always numerically stable.

Discussing the stability of the inverting polynomial, P_n, we notice that this recurrence (2.22) is neither symmetric nor homogeneous

$$y_n - a_n K y_{n-1} - (1-a_n)y_{n-2} = a_n. \tag{2.38}$$

[5]For the sake of simplicity, a_n is replaced by its limit A (cf. (2.25)).

We obtain a homogeneous relation by shifting $n \to n+1$ and subtract (2.38). Again we substitute a_n by its limit A and impose $y_n = \lambda^n$

$$\begin{aligned} &\lambda^{n-2}\left[\lambda^3 - (1+AK)\,\lambda^2 - ((1-A) - AK)\,\lambda + (1-A)\right] = 0 \\ &\lambda^{n-2}\left[\left(\lambda^2 - AK\lambda - (1-A)\right)\cdot(\lambda - 1)\right] = 0. \end{aligned} \tag{2.39}$$

From (2.39) we can read off $\lambda_0 = 1$ and solving the quadratic part yields $\lambda_{1/2} = \frac{AK}{2} \pm \sqrt{A^2K^2/4 + (1-A)}$. The general solution of (2.38) is thus the linear combination

$$y_n = c_0 + c_1\lambda_1^n + c_2\lambda_2^n. \tag{2.40}$$

From the initial values $y_0 = 1$ and $y_1 = A(1+K)$ we obtain expressions for

$$c_{1/2} = \pm\left(A(1+K) - c_0(1-\lambda_{2/1}) - \lambda_{2/1}\right)/(\lambda_1 - \lambda_2). \tag{2.41}$$

Apparently, the determination of c_0 is quite involved. However, c_0 is irrelevant when analyzing the stability of the recursion since it does not enter with a power of n. When taking the limit of $n \to \infty$, we observe as before that always one solution is growing and one decaying. Both are linearly combined and we conclude that also this recursion is always stable.

Chapter 3

Even-Odd Preconditioning

After integrating out the anti-commuting Grassmann variables it remains to compute the determinant of the fermion matrix

$$\det\{M^\dagger(U)M(U)\}. \tag{3.1}$$

Here $M^\dagger M$ is considered to ensure positivity and interpreted as two flavors. If all sites x of the lattice are labelled either as *even* or *odd* according to the sum of its coordinates, $\sum_{\mu=0}^{3} x_\mu$, being even or odd we can apply *even-odd preconditioning*. Reorganizing the fermion matrix accordingly we achieve

$$M = \begin{bmatrix} \mathrm{M_{ee}} & \mathrm{M_{eo}} \\ \mathrm{M_{oe}} & \mathrm{M_{oo}} \end{bmatrix}, \tag{3.2}$$

where the subscripts e and o refer to the *even* and *odd* sites, respectively. The even-even and odd-odd components are given by the shifted Sheikholeslami-Wohlert term, while the even-odd and odd-even contributions are given by the hopping operator. Hence $\mathrm{M_{ee}}$ and $\mathrm{M_{oo}}$ are Hermitian, whereas for $\mathrm{M_{eo}}$ and $\mathrm{M_{oe}}$ holds ${\mathrm{M_{eo}}}^\dagger = \gamma_5\,\mathrm{M_{oe}}\,\gamma_5$. Spectral properties of the hopping operator and the effect of the improvement terms are discussed in detail in the following chapter.

After reorganizing the lattice into even and odd sites it is favorable to store the lattice in the same fashion. This enhances the computation because a faster memory access becomes possible. When computing e.g. the hopping contributions of the *even* sites only *odd* sites are accessed.

More important than this technical speed-up is the algorithmic gain which is achieved if the determinant is factorized. Here two different versions are used which commonly are denoted as *asymmetric* and *symmetric* even-odd preconditioning.[32–34] Subsequently both will be derived.

3.1 Asymmetric Version

Factorizing expression (3.2) to yield the asymmetric version either $\mathrm{M_{ee}}$ or $\mathrm{M_{oo}}$ are factorized

$$M = \begin{bmatrix} \mathrm{M_{ee}} & 0 \\ \mathrm{M_{oe}} & \mathbb{1} \end{bmatrix} \cdot \begin{bmatrix} \mathbb{1} & \mathrm{M_{ee}^{-1}\,M_{eo}} \\ 0 & (\mathrm{M_{oo}} - \mathrm{M_{oe}\,M_{ee}^{-1}\,M_{eo}}) \end{bmatrix}. \tag{3.3}$$

Here we select the even sites but by interchanging all even labels with the odd ones and odd with even sites the discussion remains entirely general. Computing next the determinant of M we get

$$\det\{M\} = \det\{\mathrm{M_{ee}}\} \cdot \det\{\hat{\mathrm{M}}^{\mathrm{A}}\} \tag{3.4}$$

and thus computing the determinant of

$$\hat{\mathrm{M}}^{\mathrm{A}} = \mathrm{M_{oo}} - \mathrm{M_{oe}\,M_{ee}^{-1}\,M_{eo}} \tag{3.5}$$

becomes the challenge. $\hat{\mathrm{M}}^{\mathrm{A}}$ is non-Hermitian and using the properties of M_{xy} with $x, y = \{e, o\}$ we find using $[\gamma_5, \mathrm{M_{ee}}] = [\gamma_5, \mathrm{M_{oo}}] = 0$

$$\hat{\mathrm{M}}^{\mathrm{A}\dagger} = \gamma_5\,\hat{\mathrm{M}}^{\mathrm{A}}\,\gamma_5. \tag{3.6}$$

After even-odd preconditioning the partition function for two dynamical flavors reads

$$\mathcal{Z} \propto \int \mathcal{D}U\; e^{-S_G(U)} \cdot \det\{\hat{\mathrm{M}}^{\mathrm{A}\dagger}(U)\,\hat{\mathrm{M}}^{\mathrm{A}}(U)\} \cdot \det\{\mathrm{M_{ee}}^2(U)\}. \tag{3.7}$$

Since the determinant is invariant under multiplying γ_5 we can alternatively define the Hermitian operator[1]

$$\hat{Q}_A = \gamma_5 \hat{M}_A = \gamma_5 \left(\mathrm{M_{oo}} - \mathrm{M_{oe}\,M_{ee}^{-1}\,M_{eo}}\right) \tag{3.8}$$

representing the fermionic degrees of freedom.

3.2 Symmetric Version

To achieve the symmetric version of even-odd preconditioning both $\mathrm{M_{ee}}$ and $\mathrm{M_{oo}}$ are factorized

$$M = \begin{bmatrix} \mathrm{M_{ee}} & 0 \\ \mathrm{M_{oe}} & \mathbb{1} \end{bmatrix} \cdot \begin{bmatrix} \mathbb{1} & (\mathrm{M_{ee}^{-1}} - \mathbb{1})\,\mathrm{M_{eo}\,M_{oo}^{-1}} \\ 0 & (\mathbb{1} - \mathrm{M_{oo}^{-1}\,M_{oe}\,M_{ee}^{-1}\,M_{eo}}) \end{bmatrix} \cdot \begin{bmatrix} \mathbb{1} & \mathrm{M_{eo}} \\ 0 & \mathrm{M_{oo}} \end{bmatrix}, \tag{3.9}$$

[1] Commonly, the preconditioned Hermitian operator is defined multiplying $\hat{c}_0 = (1 + 64\kappa^2)^{-1}$ for which e.g. in the force computation is accounted by multiplying its inverse. Here and if not differently stated we assume $\hat{c}_0 = 1$ and hence drop it.

leading to three factors when computing the determinant

$$\det M = \det\{\mathrm{M_{ee}}\} \cdot \det\{\mathrm{M_{oo}}\} \cdot \det\{\hat{\mathrm{M}}^{\mathrm{S}}\}. \tag{3.10}$$

Here the computation of

$$\hat{\mathrm{M}}^{\mathrm{S}} = \mathbb{1} - \mathrm{M_{oo}^{-1}}\,\mathrm{M_{oe}}\,\mathrm{M_{ee}^{-1}}\,\mathrm{M_{eo}}\,. \tag{3.11}$$

is the challenge which like $\hat{\mathrm{M}}^{\mathrm{A}}$ is non-Hermitian

$$\hat{\mathrm{M}}^{\mathrm{S}\dagger} = \gamma_5\,\mathrm{M_{oo}}\,\hat{\mathrm{M}}^{\mathrm{S}}\,\mathrm{M_{oo}^{-1}}\,\gamma_5. \tag{3.12}$$

$\hat{\mathrm{M}}^{\mathrm{A}}$ and $\hat{\mathrm{M}}^{\mathrm{S}}$ obey the mutual relations

$$\hat{\mathrm{M}}^{\mathrm{A}} = \mathrm{M_{oo}}\,\hat{\mathrm{M}}^{\mathrm{S}} \qquad \text{and} \qquad \hat{\mathrm{M}}^{\mathrm{A}\dagger} = \hat{\mathrm{M}}^{\mathrm{S}\dagger}\,\mathrm{M_{oo}}\,. \tag{3.13}$$

For symmetric even-odd preconditioning the partition function for two dynamical flavors becomes

$$\begin{aligned}\mathcal{Z} \propto \int &\mathcal{D}U\; e^{-S_G(U)} \\ &\cdot \det\{\hat{\mathrm{M}}^{\mathrm{S}\dagger}(U)\,\hat{\mathrm{M}}^{\mathrm{S}}(U)\} \cdot \det\{\mathrm{M_{ee}}^2(U)\} \cdot \det\{\mathrm{M_{oo}}^2(U)\}.\end{aligned} \tag{3.14}$$

Multiplying expression (3.11) by γ_5 results in the pseudo-Hermitian operators

$$\begin{aligned}\hat{Q}^S &= \gamma_5 \left(\mathbb{1} - \mathrm{M_{oo}^{-1}}\,\mathrm{M_{oe}}\,\mathrm{M_{ee}^{-1}}\,\mathrm{M_{eo}}\right) = \mathrm{M_{oo}^{-1}}\,\hat{Q}^A; \\ \hat{Q}^{S\dagger} &= \gamma_5 \left(\mathbb{1} - \mathrm{M_{oe}}\,\mathrm{M_{ee}^{-1}}\,\mathrm{M_{eo}}\,\mathrm{M_{oo}^{-1}}\right) = \hat{Q}^A\,\mathrm{M_{oo}^{-1}} = \mathrm{M_{oo}}\,\hat{Q}^S\,\mathrm{M_{oo}^{-1}}\,.\end{aligned} \tag{3.15}$$

In the following we like to keep the discussion general and do not specify symmetric or asymmetric preconditioning. Thus a generic $\hat{M}$ ($\hat{Q}$) is used. Both versions of even-odd preconditioning is common to carry out the evolution (in our case) only on the odd sites and thus e.g. only vectors of half the length are required.[2] Finally, we express all contributions by exponentials and write the partition function as

$$\mathcal{Z} = \int \mathcal{D}U\,\mathcal{D}\phi^\dagger\,\mathcal{D}\phi\;\exp\left\{-S_G(U) - S_{\mathrm{det}}(U) - S_{\mathrm{b}}(U,\phi^\dagger,\phi)\right\} \tag{3.16}$$

with S_G given by (1.4) and

$$S_{\mathrm{det}} = \begin{cases} -2\,\mathrm{Tr}\,\ln(\mathrm{M_{ee}}(U)) & \text{(asymmetric)} \\ -2\,[\mathrm{Tr}\,\ln(\mathrm{M_{ee}}(U)) + \mathrm{Tr}\,\ln(\mathrm{M_{oo}}(U))] & \text{(symmetric)} \end{cases} \tag{3.17}$$

$$S_b = \phi^\dagger \left(\hat{M}\hat{M}^\dagger\right)^{-1} \phi. \tag{3.18}$$

[2] Without loss of generality we selected the odd-sites and assume for simplicity the size of the lattice to be in each direction divisible by two.

Chapter 4

Spectrum of the Dirac-Wilson Operator

After introducing the Dirac-Wilson operator in Section 1.3 we are now going to discuss some of its properties and derive bounds on its spectrum. We start by considering first the hopping operator only and proceed subsequently to the $O(a)$ improved operator as it is employed in dynamical fermion simulations. Since the Sheikholeslami-Wohlert-term is an improvement term its impact on the spectrum is considered to be small and some properties of the hopping operator may be approximately true for the improved operator which itself does not allow easily for such derivations. Parts of this chapter are published in [35].

4.1 The Hopping Operator

4.1.1 Properties

Let us first note some properties of the hopping operator. Re-writing expression (1.20) as

$$H_{xy} = \sum_{\mu=0}^{3} \Big(r[U_\mu(x)\delta_{x+\hat{\mu},y} + U_\mu(x-\hat{\mu})^\dagger \delta_{x-\hat{\mu},y}] - \gamma_\mu[U_\mu(x)\delta_{x+\hat{\mu},y} - U_\mu(x-\hat{\mu})^\dagger \delta_{x-\hat{\mu},y}] \Big), \tag{4.1}$$

we observe that the first bracket in (4.1) is Hermitian, while the second contribution is anti-Hermitian. H itself is in general not normal,

$$[H, H^\dagger] \neq 0 \tag{4.2}$$
$$= 2r\sum_{\mu\neq\nu}\gamma_\nu\Big[U_\mu(x)\delta_{x+\hat{\mu},y} + U_\mu(x-\hat{\mu})^\dagger\delta_{x-\hat{\mu},y}, U_\nu(x)\delta_{x+\hat{\nu},y} - U_\mu(x-\hat{\mu})^\dagger\delta_{x-\hat{\mu},y}\Big].$$

Properties of normal and non-normal matrices are discussed in Appendix A where in addition the used norms on matrices and vectors are defined (see also References [36, 37]).

4.1.2 Spectral Bound

Estimating the norm of H we obtain a bound on its spectrum. For simplicity we take from now on $r = 1$. Furthermore, we simplify the discussion and assume to be working in an infinite volume or take the boundary conditions (BC) to be periodic

$$\begin{aligned}\|H\| \leq \sum_\mu \| & U_\mu(x)\delta_{x+\hat{\mu},y} + U_\mu(x-\hat{\mu})^\dagger\delta_{x-\hat{\mu},y} \\ & -\gamma_\mu[U_\mu(x)\delta_{x+\hat{\mu},y} - U_\mu(x-\hat{\mu})^\dagger\delta_{x-\hat{\mu},y}]\|.\end{aligned} \tag{4.3}$$

The expression on the right hand side within $\|\cdot\|$ is normal. With periodic BC the eigenvalues of H are proportional to $\mathrm{e}^{-i\varphi}$, while γ_μ only contributes ± 1

$$\begin{aligned}\|H\| &\leq \sum_\mu 2 \cdot \max_\pm |\pm i\sin\varphi + \cos\varphi| \\ &\leq 4 \cdot 2 \cdot \max_\varphi |i\sin\varphi + \cos\varphi| = 8.\end{aligned} \tag{4.4}$$

From (4.4) follows the bound on the (unimproved) Wilson-Dirac operator $M = \mathbb{1} - \kappa H$

$$\|M\| \leq 1 + \|\kappa H\| = 1 + 8\kappa. \tag{4.5}$$

4.1.3 Unit gauge field

Fixing the gauge field to $U_\mu(x) = \mathbb{1}$, $\mu = 0, \ldots, 3$, H itself becomes normal and further analytic information on the spectrum can be derived. The eigenfunctions of H are plane waves

$$\psi(x) \propto u(p)\,\mathrm{e}^{-ipx} \tag{4.6}$$

with a Dirac spinor $u(p)$. Switching to the Fourier representation the hopping operator becomes

$$H \to H(p) = \sum_{\mu} (2i\gamma_\mu \sin p_\mu + 2\cos p_\mu). \tag{4.7}$$

Therefrom we obtain the eigenvalues of H

$$\lambda(H) = 2\sum_{\mu} \cos p_\mu \pm 2i\sqrt{\sum_{\mu} \sin^2 p_\mu}. \tag{4.8}$$

The last equation allows us to determine the elliptical bound on the spectrum in the complex plane. Since sine and cosine are bounded to 1 we find the real(imaginary) half axis to be smaller equal 8(4).

In fact the bound is reached for momenta with $\cos p_0 = \cos p_1 = \cos p_2 = \cos p_3$ and we have

$$|\lambda_{\max}(H)| = 8. \tag{4.9}$$

Further details e.g. the enclosed "wholes" can be derived (see e.g. [37, 38]).

Considering $M = \mathbb{1} - \kappa H$ we observe that M becomes singular for $\kappa \to \kappa_c = 1/|\lambda_{\max}(H)| = 1/8$. Hence κ_c is named κ_{critical}.

4.1.4 Semi-Analytic Spectrum in the Schrödinger Functional

With the previously found results we now focus on the spectrum of the hopping operator in the SF fixing the gauge field U_μ still to unity. Since the relevant difference between periodic BC and SF is the Dirichlet BC in time direction we start by separating space and time

$$\psi(x) = \psi(x_0) \cdot \mathrm{e}^{i\vec{p}\vec{x}}\,. \tag{4.10}$$

As before the eigenvalue problem for the spatial components can be computed analytically (cf. eq. 4.8). Thus we reduced the problem to find the eigenvalues of the effective one-dimensional operator

$$H^{\text{eff}} = \frac{1+\gamma_0}{2}\,\mathrm{h}_0^\dagger + \frac{1-\gamma_0}{2}\,\mathrm{h}_0 + i\gamma_1\alpha, \tag{4.11}$$

where the nilpotent hopping operator in time direction acting on $\psi(x_0)$ reads

$$\mathrm{h}_0 = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & \ddots & & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 0 & & \dots & 0 & 1 \\ 0 & & \dots & 0 & 0 \end{pmatrix}. \tag{4.12}$$

Equivalently, we may find the $2(T-1)$ roots of

$$0 = \det[H^{\text{eff}} - \lambda] = \det[\lambda^2 + \lambda(\mathrm{h}_0 + \mathrm{h}_0^\dagger) + \mathrm{h}_0\,\mathrm{h}_0^\dagger + \alpha^2] \tag{4.13}$$

$$\text{with} \quad \alpha^2 = \sum_{k=1}^{3} \sin^2(p_k). \tag{4.14}$$

Eq. (4.13) can not be solved in closed form, unfortunately. Computing the eigenvalues of H^{eff} numerically for some range of α, we plot these eigenvalues in Figure 4.1 together with the ones corresponding to (anti)periodic boundary conditions which follow trivially from Fourier expansion.

For small α the eigenvalues in case of periodic BC approach 1 leading to zero-modes in $\mathbb{1} - \kappa_c H$. Whereas in case of the SF the eigenvalues are 'deflected' away from unity as $\alpha \to 0$ and show the behavior

$$|\lambda| \propto \alpha^{1/T}, \tag{4.15}$$

which leads to a gap of order $1/T$ for the SF, maintained even in the continuum limit.

Adding temporal and spatial contributions we are enabled to compute the full spectrum of H in the SF numerically

$$\text{spec}(H) = 2\,\text{spec}(H^{\text{eff}}) + 2\sum_{k=1}^{3} \cos(p_k). \tag{4.16}$$

The result for a 16^4 lattice is shown in Figure 4.2 where the elliptical bound is given by an ellipse with major half axis $a = 7.971$, minor half axis $b = 3.932$

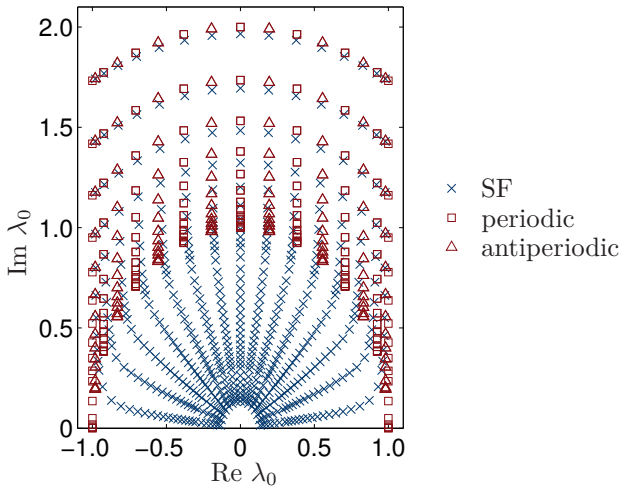

Figure 4.1. Numerical spectrum of H^{eff} for $T = 16$ and $\alpha^2 = \text{tiny} \ldots 3$.

and eccentricity $e = \sqrt{a^2 - b^2} = 6.933$. In the centers of the void areas there are (degenerate) eigenvalues for which α vanishes due to the zero mode of h_0. Roundoff errors in the eigenvalue routine in combination with the behavior (4.15) 'inflate' three of these dots to small circles.

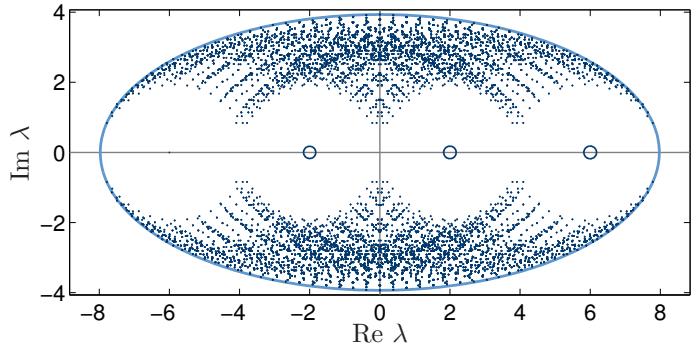

Figure 4.2. Spectrum of the hopping operator on a 16^4 lattice with $U_\mu \equiv \mathbb{1}$, $\mu = 0, \ldots, 3$ and SF boundary conditions.

4.1.5 Effect of Even-Odd-Preconditioning

As is discussed in Chapter 3, a well known trick to enhance computations is even-odd preconditioning. Neglecting the Sheikholeslami-Wohlert term by setting c_{sw} to 0 the distinction between the asymmetric and symmetric version vanishes since $\mathrm{M_{ee}} = \mathrm{M_{oo}} = \mathbb{1}$ and the preconditioned operator becomes $\hat{M} = 1 - \mathrm{M_{oe}}\,\mathrm{M_{eo}} = 1 - \hat{K}$, with $\hat{K} = \mathrm{M_{oe}}\,\mathrm{M_{eo}}$. In this limit the spectrum of K obeys an exact mapping to the spectrum of $\hat{K}$ since

$$\det\{\lambda - K\} = \det \begin{bmatrix} \lambda & -\mathrm{M_{eo}} \\ -\mathrm{M_{oe}} & \lambda \end{bmatrix} = \lambda^2 - \mathrm{M_{oe}}\,\mathrm{M_{eo}} = 0 \tag{4.17}$$

and

$$\det\{\hat{\lambda} - \hat{K}\} = \hat{\lambda} - \mathrm{M_{oe}}\,\mathrm{M_{eo}} = 0 \tag{4.18}$$

leading to

$$\hat{\lambda} = \lambda^2. \tag{4.19}$$

Despite this nice mapping of eigenvalues from the unpreconditioned operator to the preconditioned one, the spectrum becomes shifted and deformed as one can see e.g. by looking at Fig. 4.3.

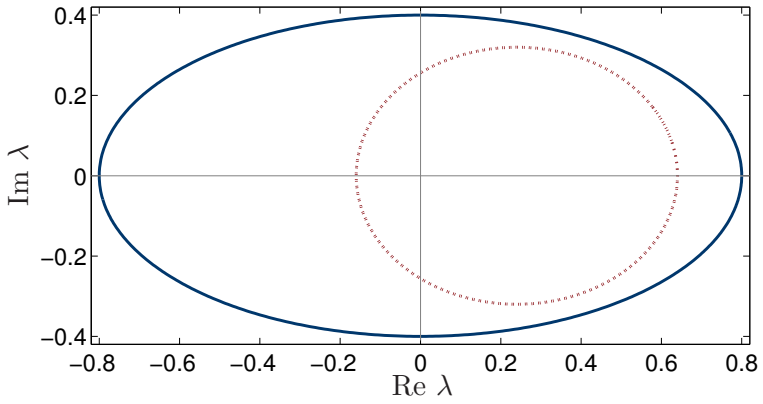

Figure 4.3. Schematic mapping of the spectral boundary of K (solid line) to the one of $\hat{K}$ (dotted line).

If we parameterize the elliptical disk containing the eigenvalues of K by $\lambda = e\cosh(\vartheta + i\varphi)$ with eccentricity e and "angles" ϑ and φ, we can derive properties of the spectrum of the preconditioned operator

$$\hat{\lambda} = e^2\cosh^2(\vartheta + i\varphi) = \frac{e^2}{2}\left(1 + \cosh 2(\vartheta + i\varphi)\right). \tag{4.20}$$

In particular we find for the eccentricity $\hat{e}$ and the shift of the center $\hat{\delta}$ (referring to the elliptical disk containing the spectrum of $\hat{M}$)

$$\hat{e} = \hat{\delta} = e^2/2, \tag{4.21}$$

while the "angle" ϑ becomes doubled and $\hat{\varphi}$ encircles twice for φ going from $0 \to 2\pi$.

Considering the $O(a)$ improved operator an exact mapping can not be established. But one may assume that an improvement term has only a little impact.

A variant of even-odd preconditioning introduces a parameter $1 < \omega < 2$ to allow for *overrelaxation* falling back to the standard even-odd preconditioning for $\omega = 1$ (relaxation).[39] Tuning ω to its optimal value the shift from the even-odd preconditioning is compensated and the spectrum fills a circular disk.[40] Since the Chebyshev approximation collapses for a spectrum on a circular disk to the geometric series (cf. next Section), we expect only little gain by combining it with overrelaxation. Therefore, overrelaxation will not be considered any further.

4.2 Approximating the inverse Dirac-Wilson Operator

Given the approximations presented in Chapter 2 we like to verify that the derived polynomials indeed approximate the inverse Dirac-Wilson operator well and how the approximation depends on the chosen bound enclosing the spectrum. Before discussing "optimal" choices we neglect the elliptical shape and consider the easier case of a circular bound and approximate the inverse of the Dirac operator ($M = \mathbb{1} - K$) by a geometric series.

4.2.1 Geometric Series

The first numerical test will be to monitor the remainder (2.7) as a sequence in n by repeatedly applying K to a random vector field η. Assuming K to be unitary diagonalizable the radius of the circular bound is given by the eigenvalue of K with largest magnitude. This value determines how fast the remainder converges. Since generally K is not unitary diagonalizable (remember: K is non-normal) this behavior is true only asymptotically and therefore non-monotonic convergence is possible. A short discussion deriving the appropriate formulae is given in Appendix A.2. As long as all eigenvalues are smaller than 1 the sequence will converge for large n. Extracting the factor κ from K, $K = \kappa \cdot \mathcal{K}$, we set $\kappa_c = 1/|\lambda_{\max}(\mathcal{K})|$. In the limit of $\kappa \to \kappa_c$ convergence breaks down and we define the rate of convergence μ^{Geo} by

$$\mu^{\text{Geo}} = -\ln\left(\frac{|\lambda_{\max}(\kappa\mathcal{K})|}{|\lambda_{\max}(\kappa_c\mathcal{K})|}\right) = -\ln(\kappa \cdot |\lambda_{\max}(\mathcal{K})|) = -\ln(\lambda_{\max}(K)). \quad (4.22)$$

The last relation in eq. (4.22) states that the convergence in case of the geometric series depends only on κ and $\lambda_{\max}(\mathcal{K})$. κ is a given parameter and $\lambda_{\max}(\mathcal{K})$ a quantity of the $O(a)$ improved hopping operator. Hence μ^{Geo} is expected to be rather insensitive to the chosen η but will be influenced by the boundary condition due to its impact on the eigenvalues of K.

4.2.2 Chebyshev Approximation

Including the fact that the spectrum of the Dirac operator has an elliptical shape, we arrive at the Chebyshev approximation and its remainder is given by eq. (2.20). Instead of exploiting the relation $M = \mathbb{1} - K$ and assuming the spectrum of K to be origin-centered, we like to keep the discussion more general and allow the spectrum of K to be shifted along the real axis by a

constant δ.[1] Then $\widetilde{K} = K + \delta$ is bounded by an origin-centered ellipse with semi major(minor) half axes a (b) defining the eccentricity $e = \sqrt{a^2 - b^2}$. The appropriate expression in terms of the scaled and translated Chebyshev polynomials [26] is

$$R_{n+1}(M) = \frac{T_{n+1}((d-M)/e)}{T_{n+1}(d/e)}. \tag{4.23}$$

which for $d = 1$, $\delta = 0$ equals (2.20). The eigenvalues $\tilde{\lambda}$ of $\widetilde{K}$ can be parameterized by

$$\tilde{\lambda} = e\cosh(\vartheta + i\varphi) = e\left[\cosh\vartheta\cos\varphi + i\sinh\vartheta\sin\varphi\right] \tag{4.24}$$

and we find the relation for the half axis

$$a = e\cosh\vartheta \quad \text{and} \quad b = e\sinh\vartheta, \tag{4.25}$$

which allows us to obtain ϑ by $\tanh\vartheta = \frac{b}{a}$. Furthermore, we note the relation $M = 1 + \delta - \widetilde{K}$, thus $d = 1 + \delta$ and $\lambda = \tilde{\lambda} - \delta$.

Next we determine the rate of convergence and assume for simplicity the matrix $\widetilde{K}$ to be unitary diagonalizable.[2] Replacing $\widetilde{K}$ by its eigenvalues and applying moreover the cosh-definition of the Chebyshev polynomials we yield

$$R_{n+1}(M) = \frac{T_{n+1}(\widetilde{K}/e)}{T_{n+1}(d/e)} = \frac{T_{n+1}(\tilde{\lambda}/e)}{T_{n+1}(d/e)} = \frac{\cosh\Big((n+1)\ \mathrm{arcosh}(\tilde{\lambda}/e)\Big)}{\cosh\left((n+1)\ \mathrm{arcosh}(d/e)\right)}. \tag{4.26}$$

From this we obtain the bound

$$|R_{n+1}(M)| \leq \frac{\cosh\left((n+1)\,\vartheta\right)}{\cosh\left((n+1)\,\alpha\right)}, \tag{4.27}$$

where we introduced $\alpha = \mathrm{arcosh}(d/e)$, which denotes the point of inversion of the ellipse. Taking now the limit of $n \to \infty$

$$\lim_{n\to\infty} |R_{n+1}(M)| \approx \exp\{-(n+1)(\alpha - \vartheta)\}. \tag{4.28}$$

we determine the rate of convergence for the Chebyshev approximation

$$\mu^{\mathrm{Cby}}(d, e, a) = (\alpha - \vartheta) = \ln\left(\frac{d + \sqrt{d^2 - e^2}}{a + b}\right) \quad \text{with } b = \sqrt{a^2 - e^2}. \tag{4.29}$$

If we consider the origin-centered ellipse ($d = 1$) which degenerates to a circle, the eccentricity vanishes, $e \to 0$, and $\mu^{\mathrm{Cby}} \to \ln(\frac{1}{a})$. The half axis a corresponds to the radius of the circle which is given by the norm of the eigenvalue of largest magnitude. Hence (4.22) is recovered.

[1] Alternatively, one could scale the entire operator by δ.

[2] In general $\widetilde{K}$ is non-normal but for n large enough the impact of the deviation from normality (A.22) becomes negligible.

How to Determine the Ellipse

In case of a free field with periodic BC we are in the favorable position that the eigenvalue with maximal real and the one with maximal imaginary component are known analytically and both lie on the real and imaginary axis, respectively. Thus it is easy to determine a and b and compute e and ϑ

$$a = 8; \quad b = 4; \quad e = \sqrt{48} \quad \tanh \vartheta = \tfrac{1}{2}. \tag{4.30}$$

Using these values μ^{Cby} becomes a function of κ only

$$\mu^{\text{Cby}}_{\text{PBC}}(\kappa) = \ln\left(\frac{1+\sqrt{1-48\kappa^2}}{12\kappa}\right) \tag{4.31}$$

and we observe that $\mu^{\text{Cby}}_{\text{PBC}} \to 0$ in the limit of $\kappa \to \kappa_c = \frac{1}{8}$.

Generally, the eigenvalue which has the maximal real component may not have a vanishing imaginary part and vice versa. Thus finding the "best fitting" ellipse becomes more challenging. Nevertheless, these eigenvalues may provide a good hint to find the "optimal" eccentricity needed as input for the Chebyshev approximation and we may also get an idea how the rate of convergence will be.

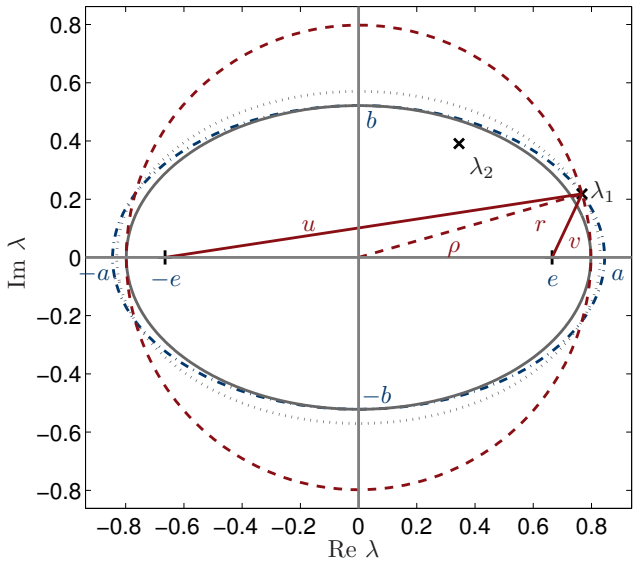

Figure 4.4. Finding ellipses bounding the largest eigenvalues of a 6^4 free field hopping operator with SF boundaries. Solid grey line initial guess, dotted grey line a bounding confocal ellipse and blue line with dash-dot pattern the bounding ellipse keeping the initial b fixed.

First we seek the eigenvalue with largest real (λ_1) and largest imaginary (λ_2) component, compute their norms

$$r_1 = \|\lambda_1\|; \qquad r_2 = \|\lambda_2\|, \tag{4.32}$$

and use these values as "initial guess" for $a = r_1$ and $b = r_2$, respectively. Therefrom we determine the eccentricity $e = \sqrt{a^2 - b^2}$. In case of a 6^4 lattice with SF boundary conditions and free field these points are plotted in Figure 4.4. The red dashed circle corresponds to the radius r of the geometric series approximating M^{-1}. The ellipse corresponding to our initial guess for a and b is drawn as grey solid line. As expected this ellipse is not a bound on the spectrum; the eigenvalue of the largest real component is not even enclosed by it. Thus we are seeking the confocal ellipse which passes through λ_1. Therefore we determine the angle ρ, which is given by the projection of r_1 onto the axis

$$\rho = \arctan(\mathrm{Im}\,\{\lambda_1\}/\mathrm{Re}\,\{\lambda_1\}). \tag{4.33}$$

Subsequently, we specify u and v using the law of cosines

$$\begin{aligned} u^2 &= e^2 + r_1^2 - 2er_1\cos(\pi - \rho) \\ v^2 &= e^2 + r_1^2 - 2er_1\cos(\rho) \end{aligned} \tag{4.34}$$

and yield by the defining equation of ellipses (2.3) the major half axis a from which together with e the minor half axis b easily follows. This ellipse provides now a bound of the spectrum (grey dotted line in Figure 4.4).

Another construction derives from the observation that the eigenvalue with largest imaginary component is described rather well by the initial guess for b. Keeping $b = r_2$ fixed we find via the parameter form (2.4) ϱ which then determines a such that the eigenvalue λ_1 is included,

$$\begin{aligned} \varrho &= \arcsin(\mathrm{Im}\,\{\lambda_1\}/b) \\ a &= \mathrm{Re}\,\{\lambda_1\}/\cos(\varrho), \end{aligned} \tag{4.35}$$

(drawn in blue with a dash-dot pattern).

Considering larger lattices the discretization of the possible momenta becomes finer and the largest eigenvalues move closer to the axis. Consequently, the initial guess improves and the choices of e will differ less.

Finding an Optimal Ellipse

A completely different approach to find the optimal ellipse is based on the assumption to have eigenvalues of the spectral boundary altogether building

up an elliptical curve. Considering these eigenvalues as set of of data-points (x_i, y_i), we can obtain the ellipse describing them best by fitting. We start from the (origin-centered) ellipse in parameter form

$$x = a\cos(t) \quad \text{and} \quad y = b\sin(t) \tag{4.36}$$

and define using the identity for ellipses the function

$$f = \frac{x^2}{a^2} + \frac{y^2}{b^2} - 1. \tag{4.37}$$

Searching numerically for the minimum of (4.37) we obtain the best-fitting ellipse centered at $(0, 0)$ with both half axis aligned to the coordinate axes.

Generalizing this ansatz a shifted ellipse centered at (x_0, y_0) can be fitted as well as a tilted one which half axis are rotated by the angle φ. A compact `Matlab` function can be found in [41].

Identifying the set of data-points with complex eigenvalues $\lambda_i = x_i + iy_i$ as they are obtained by Lanczos' method (cf. Appendix B.1), a bounding ellipse on the spectrum can be obtained by fitting. But, as discussed in the previous section, the spectral boundary can deviate from our assumption to be elliptical due to a non-trivial gauge-field, the SF boundary conditions and/or the Sheikholeslami-Wohlert term. Hence fitting the eigenvalues to an ellipse will lead to a too small ellipse, which can not serve as a bound on the spectrum. Thus we encounter a significant systematic deviation.

Therefore, we use the fitted parameters only as initial guess for finding the optimal ellipse which we define to maximize the rate of convergence μ (4.29) and encloses at least 97% of our eigenvalues.[3] Moreover, we constrain the ellipse to be (according to the previously discussed symmetries) untilted and only horizontally shifted by δ, thus $(x_0, y_0) = (\delta, 0)$. Varying now the eccentricity e and the shift δ we compute for each pair (e, δ) and for all eigenvalues λ_i the parameter ϑ_i, which follows form eq. (4.24)

$$\vartheta_i = \operatorname{arcosh}\left(\frac{\lambda_i - \delta}{e}\right). \tag{4.38}$$

A pair (e, δ) specifies a set of confocal ellipses and ϑ_i prescribes which of these ellipses passes through the eigenvalue λ_i. Cutting off a few (3%) largest values of ϑ_i we remove the outlying scatters from our analysis and compute μ for

[3] Empirically, we found 97% to be working fine, getting reliable ellipse parameters without too strong influence of scatters by the Lanczos method. The outlining points removed are not considered to be "true eigenvalues" but artefacts of the Lanczos-algorithm (cf. Appendix B.1).

the then largest value of ϑ_i. Having found the maximal μ the corresponding ellipse is centered at $(\delta, 0)$ and fully specified by its eccentricity e and ϑ allowing to determine a and b by (4.25).

4.3 Numerical Studies

The aim of our first numerical studies is to test some basic properties of the spectrum of H e.g. to guess the eccentricity and then to verify whether the approximations work as expected. Moreover we look for effects due to H being non-normal. Therefore we created two test environments: one in `Matlab` (to test on a PC with various linear algebra routines available right away) and one in `TAO` for tests on the `APE1000` parallel computer. In both we implemented the four-dimensional H operator for periodic (P) and Schrödinger Functional (SF) boundary conditions. On the spatial components we additionally introduce the phase factor $\exp\{i\theta/L\}$ which gives us periodic spatial BC for $\theta = 0$ and anti-periodic for $\theta = \pi$.

The tests reported start with the easy case of a unit gauge field. This allows us to check the code by comparing with (semi)analytically known values. Later we turn to more realistic setups by allowing a non-trivial gauge field and incorporate $O(a)$ improvement or use even-odd preconditioning. Those tests are only performed on the `APE`.

4.3.1 Unit Gauge Field

Numerical Convergence

We test the numerical convergence in the `Matlab` test environment by setting up a 6^4 lattice and implementing the free hopping operator H with $N_{\text{color}} = 1$ as sparse matrix. To guarantee convergence we choose $\kappa = 0.115 < 1/8$. Computing for $n = 1, \ldots, 100$ the powers of $(\kappa H)^n$, we monitor its Frobenius-norm (see (A.10)). Since the spectral radius of κH is smaller than 1 we expect $(\kappa H)^n$ to converge to 0 as n goes to infinity (geometric series). Thereby $\|\kappa H\|_F$ gives us a measure on the convergence. Moreover each matrix $(\kappa H)^n$ is brought to upper triangular form by `Matlab`'s Schur decomposition (cf. Appendix A.2) and split into a diagonal matrix D and the strictly upper triangular matrix N. The Frobenius norms of D and N are monitored, too.[4]

[4]Of course, if one is only interested in the numerical convergence of the geometric series there is a much cheaper version of this test being presented subsequently. But that does not allow to get hold on the non-normality of H to which we draw our attention now.

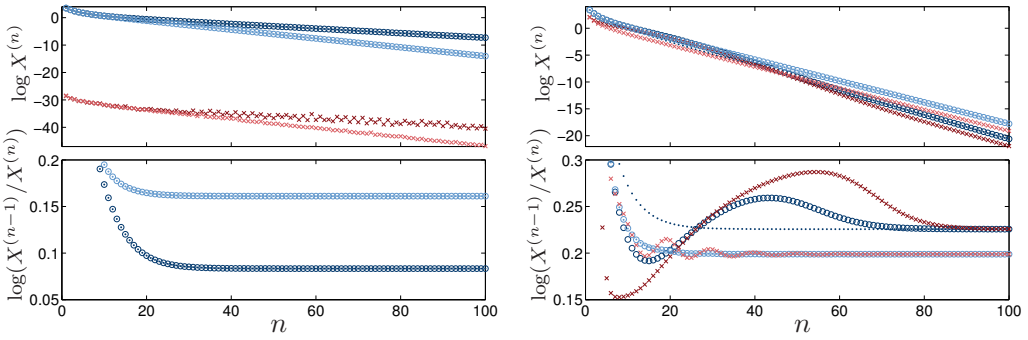

Figure 4.5. Convergence of a free field (non-)normal $(\kappa H)^n$ for $n = 1, \ldots, 100$ on a 6^4 lattice at $\kappa = 0.115$. Left: PBC (H normal); right SF (H nonnormal). A measure on the non-normality is the Forbenius-norm of the Schur decomposition of H computed for each n, where $X^{(n)}$ refers to $\circ\ \|(\kappa H)^{(n)}\|_F$; $\cdot\ \|D^{(n)}\|_F$ and $\times\ \|N^{(n)}\|_F$. Dark colors indicate $\theta = 0$, light colors denote $\theta = \pi$.

Looking first at the left, upper plot in Fig. 4.5 we find that in case of *periodic* boundary condition in time $\|H\|_F$ falls off exponentially and is equal to the norm of D while there is – up to round-off errors – no contribution from $\|N\|_F$. Plotting the differential change of the logarithm of the norms (lower plot) we observe that for $n = 40$ dark symbols and $n = 30$ light symbols the decay becomes constant. The dark symbols refer to the case of periodic spatial BC ($\theta = 0$), while the light symbols correspond to $\theta = \pi$ and thus anti-periodic spatial BC. Hence this confirms H is unitary diagonalizable and thus normal.

In the plot on the right we show the results of the test using Dirichlet boundary condition in time (SF). Focussing our attention to the upper plot one sees that the contribution of $\|N\|_F$ is no longer negligible. In case of the Schrödinger Functional where the spatial components are periodic ($\theta = 0$, dark symbols), $\|N\|_F$ becomes even larger than $\|D\|_F$ at around $n = 20$. Whereas for $\theta = \pi$ (antiperiodic spatial BC, light symbols) $\|D\|_F$ is always larger than $\|N\|_F$. This behavior becomes more explicit when plotting the data differentially (see lower plot). For Dirichlet BC in time and periodic BC in space one clearly sees that the non-normality of H affects its convergence. Until $n = 15$ the convergence of κH suffers from the contribution of N and is significantly lower than the convergence of D. For powers larger than 15 the contribution of N becomes rapidly smaller which is probably due to N being nil-potent. Hence N has to vanish for high enough powers. Then the convergence of κH raises and is from $n = 25$ to 70 larger than the convergence of D which it asymptotically approaches. κH itself becomes more and more a diagonally dominant matrix.

Focussing at the light symbols which correspond to antiperiodic BC in time we find that H is from the beginning a diagonally dominant matrix since $\|N\|_F < \|D\|_F \ \forall \ n$. This can be understood by the fact that for $\theta = \pi$ no spatial zero-modes appear and thus the non-normality is "damped". The decay of $\|\kappa H\|_F$ is similar to the one in case of periodic BC in time and becomes constant for $n = 30$. Different is the behavior of $\|N\|_F$. $\|N\|_F$ is not negligible approaching like damped oscillation the value of $\|D\|_F = \|H\|_F$.

Unfortunately, it is numerically not feasible to compute the Schur decomposition for a $SU(3)$ hopping operator of a 6^4 lattice since the computation forces a full (non-sparse) matrix and thus the available memory is exceeded.[5] Therefore, a quantitative analysis how non-normal a hopping operator on e.g. a quenched background is, could not be accomplished.

Spectral Properties

Using the implemented 4-dimensional hopping operator H we computed the eigenvalue of largest magnitude (determining the radius of the circular bound for the geometric series) as well as both the largest real and imaginary eigenvalue (required to specify the bounding ellipse for Chebyshev polynomials) using the function `eigs`. These values are known and serve mainly as consistency check.

As can be seen in Table 4.1 the values determined in the SF setup approach for larger lattices the ones of periodic boundary conditions which do not depend on the lattice size. Moreover, we notice that already for an 8^4 lattice the largest imaginary eigenvalue has a vanishing real component.

L	T	r	max(Re)	max(Im)
6	6	6.94	(6.67,1.90)	(3.00,3.40)
8	8	7.38	(7.22,1.55)	(0.00,3.86)
12	12	7.71	(7.63,1.10)	(0.00,3.91)
16	16	7.83	(7.79,0.84)	(0.00,3.93)
PBC	PBC	8.00	(8.00,0.00)	(0.00,4.00)

Table 4.1. Maximal eigenvalues of the four dimensional free field hopping operator H found with `Matlab`'s `eigs` routine.

[5] The study of a 4^4 system is refrained since the free case discussed above looks different.

Geometric Series

Testing the polynomial approximation by a geometric series we apply κH repeatedly to a random vector η drawn from a Gaussian distribution and normalized to 1. Thereby we yield r_{n+1} and compute its norm

$$r_{n+1} = \sqrt{\|R_{n+1}\eta\|^2}. \tag{4.39}$$

			PBC		SF	
T	L	θ	$\lvert\lambda_{\max}(H)\rvert$	μ^{Geo}	$\lvert\lambda_{\max}(H)\rvert$	μ^{Geo}
6	6	0	8.0000	0.0834	6.9388	0.2257
		π	7.4017	0.1611	7.1277	0.1988
8	8	0	8.0000	0.0834	7.3817	0.1638
		π	7.6589	0.1270	7.4866	0.1497
12	12	0	8.0000	0.0834	7.7114	0.1201
		π	7.8469	0.1027	7.7554	0.1144
12	6	0	8.0000	0.0834	7.1488	0.1959
		π	7.4017	0.1611	7.3388	0.1696
12	8	0	8.0000	0.0834	7.4786	0.1508
		π	7.6589	0.1270	7.5870	0.1364

Table 4.2. Rate of convergence μ^{Geo} for $T \times L^3$ lattices at $\kappa = 0.115$ determined after $n = 150$ applications of κH with unit gauge field.

The exponential decay of r_n shows the convergence of the approximation. Determining the differential change $\ln(r_n) - \ln(r_{n+1})$ we obtain a numerical estimate on μ^{Geo}. For $n = 150$ we find for all considered lattices that these values are equal to the negative logarithm of the eigenvalue of largest magnitude. The values presented in Table 4.2 refer to $T \times L^3$ lattices and $\kappa = 0.115$. Hence in these cases the convergence is dominated for $n \geq 150$ by the magnitude of the largest eigenvalue of H despite the fact that H for SF boundary conditions is non-normal

$$\mu^{\mathrm{Geo}}_{\mathrm{num}} = \ln\left(\frac{r_n}{r_{n+1}}\right) \approx \lim_{n\to\infty} \ln\left(\frac{|\lambda_{\max}(\kappa H)^n|}{|\lambda_{\max}(\kappa H)^{n+1}|}\right) = -\ln|\lambda_{\max}(\kappa H)|. \tag{4.40}$$

Chebyshev Approximation

In case of the Chebyshev approximation we first guess as discussed in Section 4.2.2 expected values of the eccentricity and predict the rate of convergence

		initial e		initial b	
L	T	e	$\mu_{\mathrm{SF}}^{\mathrm{Cby}}$	e	$\mu_{\mathrm{SF}}^{\mathrm{Cby}}$
6	6	0.604	0.249	0.665	0.246
8	8	0.723	0.191	0.789	0.178
12	12	0.764	0.161	0.796	0.162
16	16	0.779	0.155	0.797	0.158

Table 4.3. Expected eccentricity e of the operator κH and rate of convergence for the Chebyshev approximation in the Schrödinger functional at $\kappa = 0.115$.

$\mu_{\mathrm{SF}}^{\mathrm{Cby}}$ choosing $\kappa = 0.115$. e is determined by using the eigenvalues presented in Table 4.1 multiplied by κ (cf. Table 4.3).

These values are verified by computing, like before, the differential change of the remainder (4.26). R_{n+1} is implemented exploiting the recurrence relations of the Chebyshev polynomials (2.21).

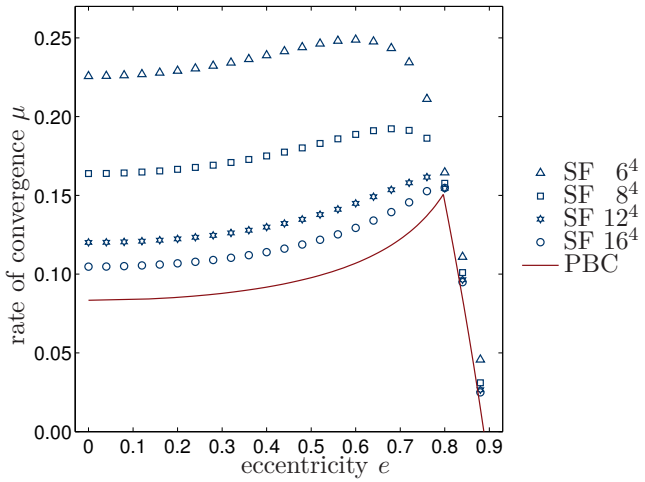

Figure 4.6. Scan over the eccentricity at $\kappa = 0.115$ for various lattice sizes with SF boundary conditions in comparison to PBC. Each point is measured after $n = 400$ iterations thus no artefacts of non-normal H are remaining.

Figure 4.6 shows these results as a scan over the eccentricity. In the Schrödinger functional μ exhibits clearly a dependence on the lattice size due to $|\lambda_{\max}|$ depending on T and L. Increasing the lattice size the convergence decreases and approaches the predicted values for periodic BC, which are independent of T and L. Moreover, we observe in all cases a hard drop-off when the eccentricity extends 0.797. Assuming the easy case of periodic boundary conditions the radius r (corresponding to $e = 0$ and approximation

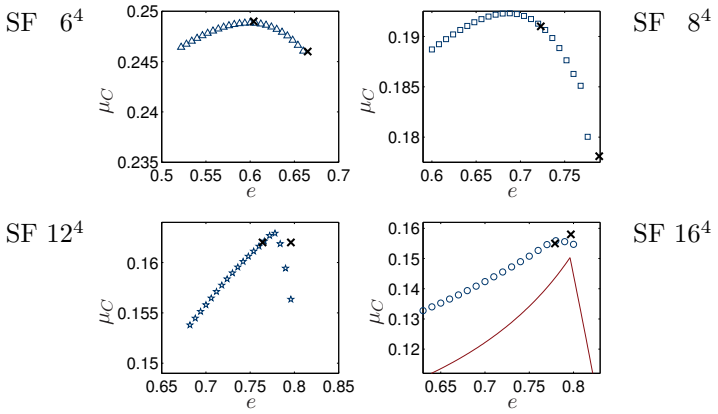

Figure 4.7. Maximal rate of convergence for various lattices at $\kappa = 0.115$. Open blue symbols denote the numerically determined value, while the black crosses indicate the predicted choices for e. The red line (16^4 lattice only) shows the theoretical expectation for a lattice with periodic BC.

with a geometric series) matches the major half axis a of the best fitting ellipse. Hence by increasing e the minor half axis b decreases while a is fixed and the convergence grows until we hit the minimum b still bounding the spectrum. Increasing e further one has to increase a. Then the bounding ellipse becomes less fitting and even extends beyond the region of convergence (negative μ^{Cby}).

Comparing our theoretical predictions (Table 4.3) with the outcome of this first scan we find the values of e are in the correct range. Therefore, a second scan focussing on the maximum of μ^{Cby} is performed and shown in Figure 4.7. The peak becomes more pronounced and moves towards the value predicted for periodic boundary conditions as the lattice size is increased. Moreover our predictions move closer to each other.

4.3.2 Dirac-Wilson Operator on Quenched Background

As a first step towards a more realistic test we replace the unit gauge field by loading a previously generated gauge field configuration. The pure gauge configurations are created on the `APE` employing a Cabbibo-Marinari heatbath update [9] combined with overrelaxation sweeps. One update cycle performs one heatbath update followed by ten overrelaxation sweeps. To yield independent configurations a gauge field is only stored after 50 update cycles.

The parameters chosen for the tests are listed in Table 4.4 and match the values previously used by the Alpha collaboration [42].

First we consider only the hopping operator H_{xy} without $O(a)$ improve-

lattice	$T \times L^3$	β	$\langle$plaquette$\rangle$	κ	c_{sw}
$P8$	8×8^3	6.00	0.63149(33)	0.13458	1.7692
$S8_a$	8×8^3	6.20	0.61037(27)	0.13458	1.6138
$S8_b$	8×8^3	6.00	0.59173(28)	0.13458	1.7692
$S8_c$	8×8^3	5.85	0.57524(34)	0.13458	2.0056
$S12$	12×12^3	6.26	0.61692(12)	0.13546	1.5827
$S16$	16×16^3	6.48	0.635128(79)	0.13541	1.4998

Table 4.4. Simulation parameters for tests using the Wilson-Dirac operator on quenched background. P periodic, S Schrödinger functional BC.

ment then we include the Sheikholeslami-Wohlert term by setting c_{sw} to its non-perturbatively determined value.

Spectral Properties

As before we start our analysis by computing the largest eigenvalues. Therefore a gauge configuration is read using the `Matlab` test environment and the eigenvalue of largest real and largest imaginary part is computed by `Matlab`'s `eigs` routine employing the Arnoldi method (cf. Appendix B.2). Unfortunately, this algorithm converged only for a subset of configurations despite the fact that the tolerance is already lowered. Hence the mean values presented in Table 4.5 are just a rough estimate and within the quoted errors no dependence on the configuration is seen.

	r	μ^{Geo}	a	b	e	μ^{Cby}
$S8_b$	0.838(4)	0.1770(7)	0.843(5)	0.47(1)	0.701(7)	0.268(3)
$P8$	0.865(5)	0.1456(8)	0.869(6)	0.48(1)	0.725(2)	0.226(9)

Table 4.5. Expected values for μ and e derived from measured maximal eigenvalues for 8^4 lattices at $\beta = 6.0$ and $\kappa = 0.13458$.

Moving to larger lattices, e.g. 12^4, this method breaks down since the `Matlab` algorithm does not converge within a reasonable amount of time. Actually, we do not need to obtain "exact" eigenvalues but are satisfied to obtain an estimate on the ellipse bounding the spectrum. For this reason we implement the Lanczos-algorithm for complex matrices as described in Appendix B.1. Computing 145 eigenvalues that way we obtain a ring of eigenvalues corresponding to the bound of the spectrum.

To show the qualitative agreement of the computed eigenvalues we plot for one 8^4 configuration 2400 eigenvalues computed by `Matlab`'s Arnoldi al-

gorithm (blue dots) and 145 eigenvalues obtained by Lanczos' method (red crosses) in Figure 4.8. Obviously, the complex Lanczos method leads to an estimate of the bound of the spectrum although we are not computing eigenvalues with high accuracy. Taking advantage of the ring of boundary eigenvalues we determine the bounding ellipse by optimizing a fit to all 145 data points of all 50 configurations as explained in Section 4.2.2. An example for the 8^4 lattices at $\beta = 6.0$ and $\kappa = 0.13458$ is shown in Figure 4.9 and the elliptical parameters derived are summarized in Table 4.6. The values are a rough estimation (leading to a lower bound) on the rate of convergence μ.

Focussing our attention on the ring of boundary eigenvalues one notices that without $O(a)$ improvement (Figure 4.9 left hand side) the shown bound on the spectrum exhibits the even-odd-symmetry under sign flip and the eigenvalues come in complex pairs due to the γ_5-Hermiticity. Moreover, we notice a deviation of the bound from an ellipse which is a lattice artefact and receding for larger lattices. The ellipse fitted to these data points is drawn as dashed red line (initial guess) and as light-blue solid line the optimal ellipse as defined in Subsection 4.2.2.

The figure on the right hand side corresponds to the same set of configurations computing this time the eigenvalues of the hopping operator with Sheikholeslami-Wohlert term. c_{sw} is set to the non-perturbative value reported in [43]. Clearly, including the clover-term the spectral bound does not exhibit the even-odd-symmetry as expected. Since both contributions obey the γ_5-Hermiticity the complex pairs of eigenvalues are preserved. Furthermore, the spectrum becomes stretched along the real axis resulting in larger values of the eccentricity than without clover-term. A tiny gain may be obtained by including the shift of the center using the parameter δ (cf. even-odd-preconditioning) which we do not exploit here.

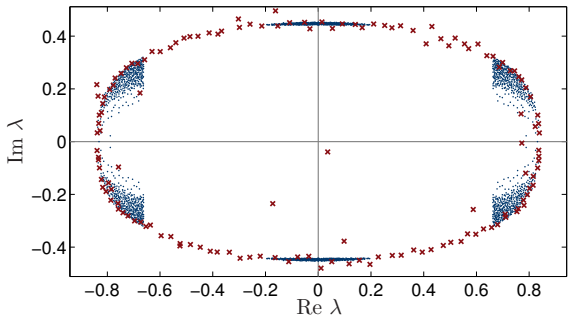

Figure 4.8. Comparing 2400 eigenvalues obtained by `Matlab`'s Arnoldi algorithm (blue dots, tolerance 10^{-2}) with 145 eigenvalues computed by Lanczos' method (red crosses) on a 8^4 gauge configuration at $\beta = 6.0$, $\kappa = 0.13458$.

lattice	Lanczos eigenvalue a	b	μ^{Geo}	δ	e	$\mu^{\text{Cby}}(e,a)$	remainder μ^{Geo}	e^{opt}	μ^{Cby}
$P8$	0.8745	0.5663	0.134	0.0000	0.6670	0.191	0.1531(16)	0.667	0.20723(96)
$S8_a$	0.8672	0.5809	0.142	0.0003	0.6439	0.198	0.1688(20)	0.650	0.2077(16)
$S8_b$	0.8550	0.5791	0.157	0.0009	0.6291	0.216	0.1835(15)	0.640	0.22246(11)
$S8_c$	0.8438	0.5824	0.170	0.0006	0.6107	0.229	0.1967(12)	0.620	0.2395(16)
$S12$	0.8900	0.5831	0.117	0.0010	0.6724	0.168	0.1370(13)	0.680	0.18147(98)
$S16$	0.9070	0.5709	0.098	0.0003	0.7048	0.146	0.11494(60)	0.725	0.16333(28)

lattice	Lanczos eigenvalue a	b	μ^{Geo}	δ	e	$\mu^{\text{Cby}}(e,a)$	remainder μ^{Geo}	e^{opt}	μ^{Cby}
$P8$	0.9957	0.7458	0.004	0.0103	0.6596	0.019	0.0107(20)	0.726	0.0119(28)
$S8_a$	0.9647	0.6366	0.036	0.0068	0.7248	0.063	0.0483(21)	0.725	0.0588(28)
$S8_b$	0.9729	0.6480	0.027	0.0078	0.7257	0.052	0.0428(13)	0.757	0.0441(23)
$S8_c$	0.9381	0.9381	0.064	0.0428	0.0003	0.106	0.0181(25)	0.000	0.0181(25)
$S12$	0.9881	0.6843	0.012	0.0078	0.7128	0.028	0.0219(20)	0.713	0.0253(21)
$S16$	0.9900	0.7089	0.010	0.0047	0.6910	0.021	0.0154(11)	0.705	0.0176(12)

Table 4.6. Upper table "pure" hopping operator, lower table $O(a)$ improved Dirac-Wilson operator.
Since δ as obtained by the Lanczos method is almost zero it is neglected in the remainder computations.

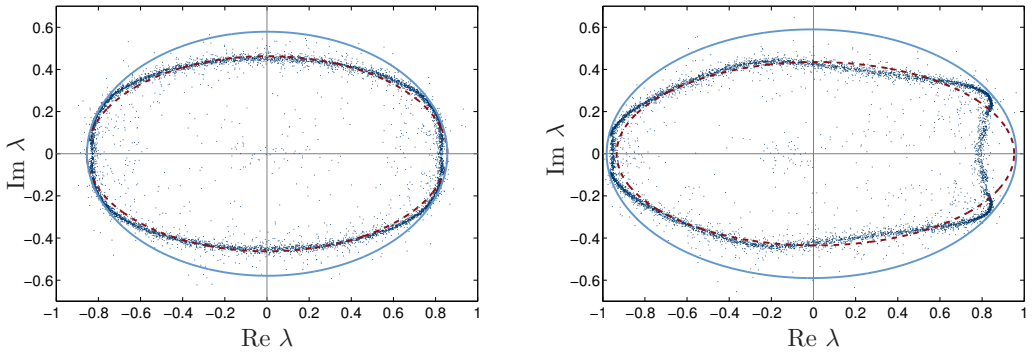

Figure 4.9. Computing 145 eigenvalues of K on 50 pure-gauge configurations using Lanczos' method. 8^4 lattice at $\beta = 6.0$ and $\kappa = 0.13458$ with SF boundary conditions. Left without / right with clover term.

Approximation Tests

Like in the case of the free field we test our predictions on the convergence and also on the elliptical shape of the spectrum by calculating the remainder R_{n+1} and monitoring its norm. We performed this test for all 50 configurations using three different random vectors to yield some statistics on μ. Moreover, we tested a set of eccentricities chosen around the value predicted by the Lanczos method. Computing μ's average and standard deviation we show the results in Table 4.6, where only the eccentricity of largest convergence (e^{opt}) is shown. Since the results of the Lanczos method confirms the spectrum to be almost centered no extra shift δ is encountered for the remainder tests.

Looking first at the data without $O(a)$ improvement we find rather good agreement between the values predicted by the Lanczos method and the ones found numerically by the remainder test. In all cases considered we find the predicted μ is a lower bound and the predicted choice for the eccentricity is smaller but close to the optimal value. Hence here we conclude our method is working and remembering the hard drop off (cf. Fig. 4.6) we conclude our predicted value for e with fixed $\delta = 0$ can be safely used.

Turning to the data with $O(a)$ improvement the situation is less appealing. First of all the convergence rates drop by an order of magnitude and the remainder is hardly converging, like in case of lattice $S8_c$.[6] Hence all predicted values are much more uncertain and the deviations between the values predicted by the Lanczos method and the results of the remainder test are large. Tiny changes in the shape of the ellipse lead to significant changes in μ. All in all there seems to be little hope to profit much using Chebyshev polynomials approximating the unpreconditioned $O(a)$ improved

[6] Incorporating a shift $\delta \neq 0$ may lead to a small improvement.

operator, also if including $\delta \neq 0$.

4.3.3 Even-Odd Preconditioning

Introducing even-odd preconditioning as described in Chapter 3, we repeat the above explained analysis and incorporate the shift $\hat{\delta}$ along the real axis when approximating M^{-1}.[7] We look at the three case:

I without Sheikholeslami-Wohlert improvement term
II with $O(a)$ improvement, factorizing $\mathrm{M_{ee}}$ only (asymmetric)
III with $O(a)$ improvement, factorizing $\mathrm{M_{ee}}$ and $\mathrm{M_{oo}}$ (symmetric)

For each case we plot the spectral boundary computed by the Lanczos method together with its corresponding bounding ellipse (see Figures 4.10 and 4.11) for the 8^4 lattice at $\beta = 6.0$ and $\kappa = 0.13458$. One sees clearly the centers of all spectra are shifted towards positive real values according to our expectations derived from eq. (4.19). Furthermore, one finds all bounding ellipses fit much tighter to the spectral bound than without preconditioning (cf. Fig. 4.9). It seems like even-odd preconditioning is curing some of the artefacts caused by the $O(a)$ improvement.

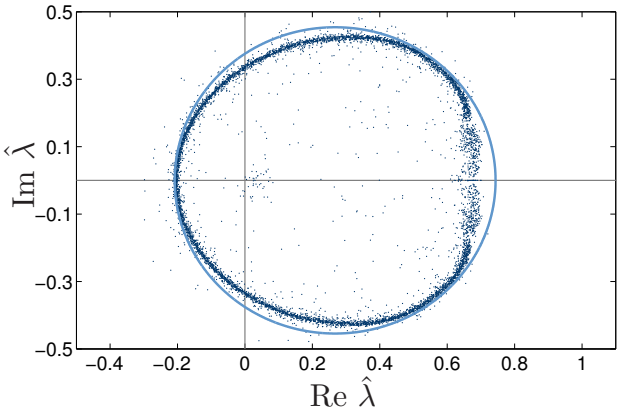

Figure 4.10. Bound of the even-odd preconditioned spectrum without Sheikholeslami-Wohlert term computed by Lanczos's method and showing its bounding ellipse. 8^4 lattice at $\beta = 6.0$ and $\kappa = 0.13458$.

Focussing our attention at the two different versions of preconditioning (Fig. 4.11) we observe that the symmetric version (III) leads to a significantly "rounder" spectrum (blue dots) than the asymmetric version (II), which in addition has a little tail at the left end of the spectrum (red dots).

[7] The modified recursion formulae are collected in Chapter 6, eqs. (6.7)-(6.9).

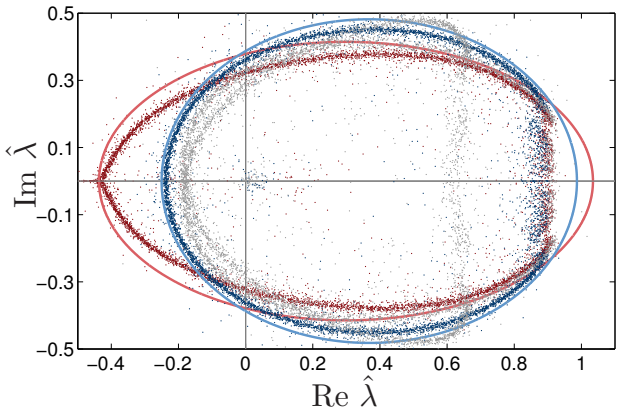

Figure 4.11. Computing the bound of the even-odd preconditioned spectrum by the Lanczos method and showing its bounding ellipse on an 8^4 lattice at $\beta = 6.0$ and $\kappa = 0.13458$ with Sheikholeslami-Wohlert term. Asymmetric version is plotted in red, symmetric in blue and in grey we exploit the mapping relation (4.19) squaring the non-preconditioned data.

Moreover we show in that plot how the Sheikholeslami-Wohlert term manipulates the mapping relation (4.19) between the unpreconditioned and the preconditioned operator (grey dots). This relation holds exactly without $O(a)$ improvement and in that case we find indeed that the squared eigenvalues of the unpreconditioned operator lie right on top of the eigenvalues of the even-odd-preconditioned operator. (Hence not plotted in Fig. 4.10.)

Next we turn to the numerical data resulting from the Lanczos method collected in Table 4.7 and check quantitatively the consequence (4.21) derived form the mapping relation (4.19). Since we estimate the optimal ellipse independently for different data sets and we use the assumption of a spectrum contained in an elliptical disk, this relation can only be approximately true. Between $\hat{e}$ and $e^2/2$ we find only poor agreement already for the case without $O(a)$ improvement (cf. Table 4.7, I). Moreover, $\hat{e}$ differs clearly form $\hat{\delta}$. With improvement the mapping relation itself holds only approximately for both considered versions of preconditioning. Looking for a relation between the preconditioned quantities $\hat{e}$ and $\hat{\delta}$ the best agreement of all three cases is found for symmetric even-odd preconditioning (III).

To test the prediction by the Lanczos method on how to choose polynomial parameters we repeat the remainder test. With even-odd preconditioning the horizontal shift $\hat{\delta}$ is no longer negligible. Hence we use the value of $\hat{\delta}$ as determined by the Lanczos method as fixed input for the remainder test and vary only $\hat{e}$ to find numerically the optimal value giving the largest rate of convergence. This same $\hat{\delta}$ is also used for the simpler approximation

by a geometric series ($e = 0$) and in Fig. 4.12. There we extended this test to a broad range of eccentricities to see how μ changes in dependence of e for lattices $S8_b$ and $S16$. Moreover, the functional dependence derived by eq. (4.29) is shown as dashed (solid) line.

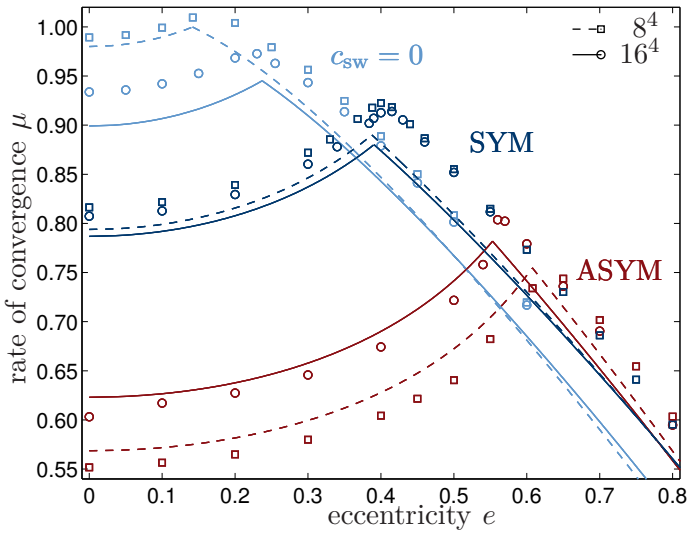

Figure 4.12. Dependence of μ on the eccentricity e for even-odd preconditioned operator on lattice $S8_b$ and $S16$ comparing theoretical prediction by the Lanczos method with numerical found values by the remainder test.

As can be seen in the plot, the general behavior of μ on e looks similar to the one shown for the free case in Fig. 4.6 but the hard drop off becomes for an operator on quenched background much weaker and the optimal value moves towards smaller eccentricities. In general the agreement between the predictions by eq. (4.29) and the numerical results obtained by the remainder test looks good. Nevertheless there are differences between our three considered cases. Without clover-term (I) the convergence rate is larger for the smaller lattice than for the larger one, while in case II (asymmetric preconditioning) this is reversed and for symmetric preconditioning (III) only small differences are seen for the 8^4 and the 16^4 lattice. Further we see that in case I and III the prediction gives always a safe lower bound on the eccentricity e to be chosen and the rate of convergence μ to be expected. Unfortunately, that is not true for the case of asymmetric preconditioning. Here it seems to be – at least partially – the little tail, which is not included in the bounding ellipse, causing the prediction to be too large. Comparing symmetric with asymmetric preconditioning the first one is clearly superior as could already be guessed from the plot of the spectral boundary.

An overview of the results of the Lanczos method in comparison with the ones from the remainder test for all in Table 4.4 considered lattices is given in Table 4.7. These data fit nicely into the picture discussed above. Hence we conclude our method to determine the approximation parameters by the Lanczos method works and leads to a rather safe choice on e and a reasonable estimate on μ. Concerning the two different versions of preconditioning the Lanczos method serves as visualization to explain the advantages of symmetric even-odd preconditioning. Numerically we find that the symmetric version leads to a gain in μ^{Cby} of roughly 10% compared to the asymmetric version and should hence be the choice to select.

The studies of the spectrum and tests to monitor the convergence of the remainder indicate that approximating the non-Hermitian $O(a)$ improved Wilson-Dirac operator with Chebyshev polynomials works but even-odd preconditioning or maybe also other forms of preconditioning are essential. Moreover, the symmetric version of even-odd preconditioning is clearly superior to the asymmetric version. Due to the Lanczos method we have also a nice tool to determine parameters relevant for the approximation.

		Lanczos eigenvalue						remainder			
	lattice	$e^2/2$	$\hat{a}$	$\hat{b}$	$\hat{\delta}$	$\hat{\mu}^{\mathrm{Geo}}$	$\hat{e}$	$\hat{\mu}^{\mathrm{Cby}}$	$\hat{\delta}$	$\hat{\mu}^{\mathrm{Geo}}$	$\hat{e}^{\mathrm{opt}}$	$\hat{\mu}^{\mathrm{Cby}}$
I	$S8_a$	0.207	0.4949	0.4518	0.2847	0.954	0.2019	0.992	0.285	0.9685(11)	0.201	1.0077(8)
	$S8_b$	0.198	0.4759	0.4543	0.2676	0.980	0.1417	0.999	0.268	0.98935(98)	0.142	1.0095(10)
	$S8_c$	0.187	0.4659	0.4503	0.2552	0.991	0.1197	1.006	0.255	1.0052(12)	0.120	1.0196(9)
	$S12$	0.226	0.5134	0.4661	0.2946	0.925	0.2154	0.965	0.295	0.94338(45)	0.215	0.9874(3)
	$S16$	0.205	0.5294	0.4733	0.3096	0.906	0.2371	0.952	0.301	0.96959(36)	0.237	0.96958(51)

		Lanczos eigenvalue						remainder			
	lattice	$e^2/2$	$\hat{a}$	$\hat{b}$	$\hat{\delta}$	$\hat{\mu}^{\mathrm{Geo}}$	$\hat{e}$	$\hat{\mu}^{\mathrm{Cby}}$	$\hat{\delta}$	$\hat{\mu}^{\mathrm{Geo}}$	$\hat{e}^{\mathrm{opt}}$	$\hat{\mu}^{\mathrm{Cby}}$
II	$S8_a$	0.263	0.7057	0.4202	0.3135	0.621	0.5669	0.797	0.314	0.6007(52)	0.600	0.7980(12)
	$S8_b$	0.263	0.7356	0.4147	0.2989	0.569	0.6075	0.755	0.299	0.5517(49)	0.650	0.7438(15)
	$S8_c$	–	0.8241	0.4020	0.3516	0.495	0.7194	0.679	0.352	0.4361(75)	0.800	0.6672(12)
	$S12$	0.254	0.7233	0.4383	0.3195	0.601	0.5754	0.769	0.320	0.5821(36)	0.625	0.76430(51)
	$S16$	0.239	0.7111	0.4470	0.3295	0.626	0.5530	0.782	0.326	0.6032(42)	0.560	0.80816(96)

		Lanczos eigenvalue						remainder			
	lattice	$e^2/2$	$\hat{a}$	$\hat{b}$	$\hat{\delta}$	$\hat{\mu}^{\mathrm{Geo}}$	$\hat{e}$	$\hat{\mu}^{\mathrm{Cby}}$	$\hat{\delta}$	$\hat{\mu}^{\mathrm{Geo}}$	$\hat{e}^{\mathrm{opt}}$	$\hat{\mu}^{\mathrm{Cby}}$
III	$S8_a$	0.263	0.6031	0.4779	0.3625	0.815	0.3678	0.906	0.363	0.8329(13)	0.380	0.93390(68)
	$S8_b$	0.263	0.6185	0.4820	0.3682	0.794	0.3877	0.890	0.368	0.8163(13)	0.400	0.92218(66)
	$S8_c$	–	0.6480	0.4964	0.3869	0.761	0.4164	0.862	0.367	0.7806(12)	0.425	0.89029(63)
	$S12$	0.254	0.6287	0.4865	0.3771	0.784	0.3981	0.883	0.377	0.80975(51)	0.415	0.91465(20)
	$S16$	0.239	0.6301	0.4945	0.3834	0.786	0.3905	0.880	0.384	0.80742(32)	0.415	0.91390(14)

Table 4.7. Results with even-odd preconditioning: I without, II / III with $O(a)$ improvement (asymmetric / symmetric version).

Chapter 5

Hybrid Monte Carlo

Combining the Monte Carlo method by Metropolis et al. [44] with the concept of molecular dynamics, Duane et al. proposed the *Hybrid Monte Carlo* (HMC) algorithm as update for simulating lattice QCD with dynamical fermions.[45] In this chapter we first introduce the two main ingredients of HMC before presenting the most widely used update-algorithm as well as some of its variants and improvements.

5.1 Metropolis' Monte Carlo

Facing the problem to sample configurations of particles in a square with very small Boltzmann factor $\exp\{-E/kT\}$ Metropolis et al. [44] proposed to

- Place N particles in any configuration e.g. a regular lattice
- Move each particle of the configuration according to $X \to X + \alpha\eta_x$ and $Y \to Y + \alpha\eta_y$, where η_x, η_y are a random numbers in $[-1, 1]$ and α is the maximal displacement
- Accept the move if ΔE, the difference of the energy between new and old configuration, is negative or for $\Delta E > 0$ accept the move according to the probability $\exp\{-\Delta E/kT\}$, i.e. a random number η in $[0, 1]$ is drawn and the move accepted if $\exp\{-\Delta E/kT\} > \eta$. For a rejected move the old configuration is repeated in the Markov chain.

The proposed method is *ergodic* because a single particle can reach for a large enough number of steps any point in the square and because this is true for all particles any point in configuration space can be reached.

5.2 Molecular Dynamics

Molecular dynamics (MD) is another approach to sample a system with desired statistical distribution.[46, 47] Adding a fifth dimension, the molecular dynamics time τ, the system is evolved along a trajectory being described by "classical mechanics". Therefore, fictitious, Gaussian momenta Π conjugate to the scalar field ϕ are introduced and the Hamiltonian

$$H(\Pi, \phi) = \frac{1}{2}\Pi^2 + S(\phi) \tag{5.1}$$

is formed, where $S(\phi)$ is the action we like to simulate. The evolution in molecular dynamics time τ is then given by

$$\dot{\phi} = \Pi \quad \text{and} \quad \dot{\Pi} = -\partial S/\partial\phi, \tag{5.2}$$

with the dot denoting the τ derivative. Hence a trajectory is defined starting from an initial configuration at $[\Pi(\tau = 0), \phi(\tau = 0)]$ to $[\Pi(\tau), \phi(\tau)]$ which has the corresponding classical partition function

$$Z = \int \mathcal{D}\Pi\, \mathcal{D}\phi\ \exp\{-H(\Pi, \phi)\}. \tag{5.3}$$

The Gaussian integral over the momenta is trivial leading to a result proportional to the path integral expression for the quantum partition function (1.25). Assuming the *ergodic hypothesis* to hold a point along the classical trajectory is visited with probability $\exp\{-S(\phi)\}$. We obtain expectation values of observables by averaging over the MD trajectory

$$\langle \mathcal{O} \rangle = \frac{1}{T} \int_{\tau_0}^{\tau_0 + T} \mathcal{D}\tau\, \mathcal{O}[\phi(\tau)], \tag{5.4}$$

where τ_0 takes into account that the system has to evolve for a sufficiently large time before being equilibrated.

The MD approach faces two problems: Firstly, one assumes the ergodic hypothesis to hold but at weak gauge couplings QCD is equivalent to a system of weakly coupled oscillators and a lack of ergodicity is expected in the continuum limit.[8] Secondly, energy violations e.g. due to the numerical integration cannot be compensated. Hence a fine and slow integration is forced.

The first problem is addressed by taking advantage of properties of the Langevin algorithm which are used in the framework of *stochastic quantization*.[17] Instead of performing many steps along a single trajectory, one performs many different steps with freshly drawn initial, Gaussian momenta.

This solves the problem of ergodicity for the price of steps with randomly changing direction. Thus one traverses slower through configuration space than by the MD approach.

The advantages of both, molecular dynamics and Langevin algorithm, are combined in the refreshed MD or hybrid-classical-Langevin algorithm.[48, 49] Instead of following one MD trajectory continuously for a long time, one refreshes the Gaussian momenta after some fixed MD time but keeps ϕ unchanged. That way ergodicity is preserved and the random changes in the direction through configuration space are suppressed.

Solving the second problem leads to the hybrid Monte Carlo algorithm discussed in the subsequent section which is nowadays the workhorse for simulating dynamical fermions in lattice QCD.

5.3 The HMC algorithm

The basic idea of HMC is to use the molecular dynamics evolution as transition function to evolve from one configuration to the next one being accepted or rejected according to a generalized Metropolis step (cf. Figure 5.1).[45] For QCD a configuration is given by the gauge-field $U_\mu(x)$ consisting of a $SU(3)_\mathrm{c}$ matrix for each Dirac index μ at each lattice site x.

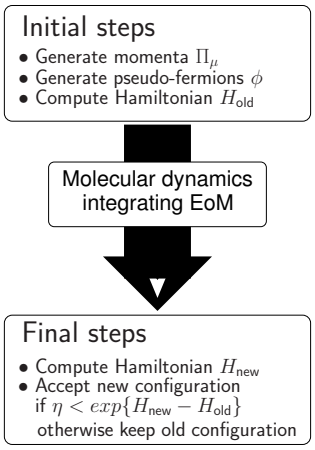

Figure 5.1. Schematic flow of the HMC algorithm.

Like in the MD approach we introduce again a fictitious Monte Carlo time τ and guide the evolution by canonical equations. To build the Hamiltonian we generate Gaussian momenta $\Pi_\mu(x)$ as conjugate variables to $U_\mu(x)$. Hence the Π_μ are traceless and Hermitian elements of the $SU(3)_\mathrm{c}$ gauge group. The

Hamiltonian is given by

$$H = \frac{1}{2}\Pi_\mu^2(x) + S_{\text{eff}}(U_\mu, \phi), \tag{5.5}$$

and evolved according to the equations of motion (EoM)

$$\dot{\phi}_\mu = \frac{\delta\{H\}}{\delta\{\Pi_\mu\}} = \Pi_\mu; \qquad \dot{\Pi}_\mu = -\frac{\delta\{H\}}{\delta\{U_\mu\}} = -\frac{\delta\{S_{\text{eff}}\}}{\delta\{U_\mu\}}. \tag{5.6}$$

The effective action S_{eff} in case of even-odd preconditioning is the sum of (1.4), (3.17) and (3.18)

$$S_{\text{eff}}(U_\mu, \phi) = S_G(U_\mu) + S_{\text{det}}(U_\mu) + S_b(U_\mu, \phi). \tag{5.7}$$

of the contributions from the gauge-field (G), the factorized part of the determinant (det) and the bosonic part (b) which in addition depends on the pseudo-fermion field ϕ.

The important difference to the MD approach comes at the end of a trajectory when the accept/reject-step is performed. Therefore we compute the energy-difference ΔH between the *new* Hamiltonian at the end and the *old* Hamiltonian at the beginning of the trajectory. The probability to accept a new configuration is given by

$$P_A((U_\mu, \Pi_\mu) \to (U'_\mu, \Pi'_\mu)) = \min\{1, \exp\{-\Delta H\}\}, \tag{5.8}$$

with $\Delta H = H_{\text{new}} - H_{\text{old}}$.

The constructed Hamiltonian serves two distinct tasks: on the one hand it enters in the accept/reject step (*acceptance Hamiltonian*), on the other hand it guides the molecular dynamics evolution (*guidance Hamiltonian*).[45] The acceptance Hamiltonian defines the equilibrium distribution to be simulated. Due to the accept/reject step the algorithm becomes exact accounting for possible rounding or discretization errors occurring e.g. by the numerical integration of the equation of motion. In principal the guidance Hamiltonian entering in the EoM can differ from the acceptance Hamiltonian allowing scope for optimization. Only if both Hamiltonian agree we find energy conservation ($\Delta H = 0$) in the limit of perfect integration.

To let the system evolve according to the EoM we have to compute the variation of the effective action (5.7) in terms of an infinitesimal change of the gauge link $\delta\{U_\mu(x)\}$

$$\delta\{S_{\text{eff}}\} = \sum_{x,\mu} \text{Tr}\, \left\{ F_\mu(x)\delta\{U_\mu(x)\} + F_\mu^\dagger(x)\delta\{U_\mu(x)^\dagger\} \right\}. \tag{5.9}$$

The quantity $F_\mu(x)$ can be considered as "force" arising for a gauge link varied. It can be split into the contributions

$$F_\mu(x) = V_\mu(x) + F_\mu^{\mathrm{det}}(x) + F_\mu^{\mathrm{b}}(x), \tag{5.10}$$

form the pure gauge part (V_μ), the determinant and the bosonic contribution.

Varying the contribution of the gauge field S_G to the action we obtain the *gauge force*

$$V_\mu = -\frac{\beta}{6} \sum_P \mathrm{Tr}\ \delta\{U_P + U_P^\dagger\}. \tag{5.11}$$

Next we focus on the *determinant force* which depends on our choice of preconditioning. Assuming both $\mathrm{M_{ee}}$ and $\mathrm{M_{oo}}$ to be factorized we are discussing the case of *symmetric* even-odd preconditioning, where S_{det} is given by

$$S_{\mathrm{det}} = -2\left[\ln\det\{\mathrm{M_{ee}}\} + \ln\det\{\mathrm{M_{oo}}\}\right]. \tag{5.12}$$

Calculating the variation of (5.12) with respect to a link U_μ we find using $\det\{A\} = \exp\{\mathrm{Tr}\ \ln A\}$

$$\begin{aligned}\delta\{S_{\mathrm{det}}\} &= -2\delta\{\mathrm{Tr}\ \ln(\mathrm{M_{ee}}) + \mathrm{Tr}\ \ln(\mathrm{M_{oo}})\} \\ &= -2\,\mathrm{Tr}\left[\mathrm{M_{ee}^{-1}}\,\delta\{\mathrm{M_{ee}}\} + \mathrm{M_{oo}^{-1}}\,\delta\{\mathrm{M_{oo}}\}\right].\end{aligned} \tag{5.13}$$

Let us now look at one particular link (pointing from x to $x+\mu$) to be varied and consider like Jansen and Liu all "clover leaves" which contain this link.[32] We find the following four diagrams for $\mu \neq \nu$ as shown in Fig. 5.2.

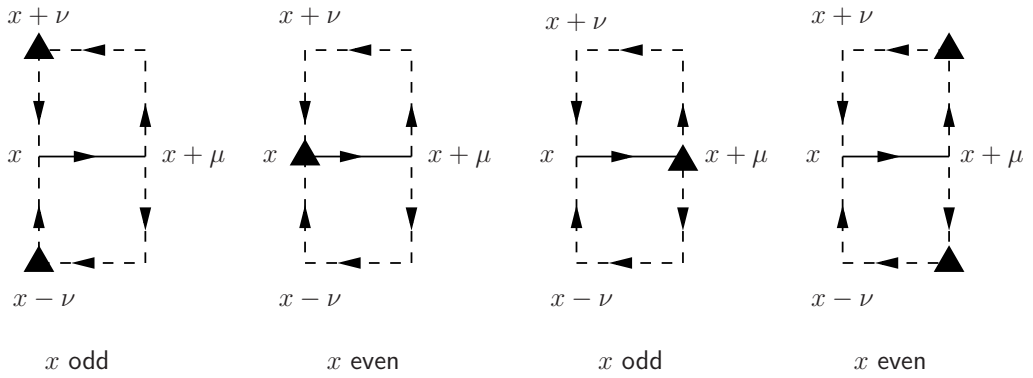

Figure 5.2. Diagrams contributing to $F_\mu^{\mathrm{det}}(x)$ with the triangles denoting "even" insertions.

To compute the contribution by varying the link $U_\mu(x)$ we have to compute the staples starting at $x+\mu$ and follow the arrows. At positions of the black triangles a (3×3)-color matrix has to be inserted which for an even insertion point is given by

$$(\blacktriangle)_x = \mathrm{Tr}_{\,\mathrm{Dirac}}\left[i\sigma_{\mu\nu}\,\mathrm{M}_{\mathrm{ee}}^{-1}(x)\right], \tag{5.14}$$

while for an odd point $\mathrm{M}_{\mathrm{ee}}^{-1}$ is replaced by $\mathrm{M}_{\mathrm{oo}}^{-1}$. Finally, we have to sum over all directions $\mu\neq\nu$ and multiply the sum by $-\kappa c_{\mathrm{sw}}/4$ i.e.

$$F_\mu^{\mathrm{det}}(x) = -\frac{\kappa c_{\mathrm{sw}}}{4}\sum_{\mu\neq\nu}(\text{diagrams in Fig. 5.2}). \tag{5.15}$$

The inversion of M_{ee} and M_{oo} is needed locally for all lattice sites x and computed e.g. exactly by applying the Householder triangularization.[36]

The last contribution, named *bosonic force*, comes from the variation of the pseudo-fermion action

$$S_b = \phi^\dagger\left(\hat{Q}\hat{Q}^\dagger\right)^{-1}\phi, \tag{5.16}$$

where the pseudo-fermion fields ϕ are generated from a Gaussian random vector η by multiplying Q to achieve a distribution according to (5.16)

$$\phi = \hat{Q}\eta. \tag{5.17}$$

Due to even-odd preconditioning $\hat{Q}$ is a mapping from odd sites on odd sites with the even sites only created/used in between. Hence η and ϕ are vectors on odd sites only. Computing the variation of (5.16) we find applying $\delta\{A^{-1}\} = -A^{-1}\delta\{A\}A^{-1}$

$$\delta\{S_b\} = -\phi^\dagger\hat{Q}^{\dagger-1}\left[\delta\{\hat{Q}^\dagger\}\hat{Q}^{\dagger-1} + \hat{Q}^{-1}\delta\{\hat{Q}\}\right]\hat{Q}^{-1}\phi. \tag{5.18}$$

The variation of the symmetric preconditioned $\hat{Q}$ is given by

$$\begin{aligned}
\delta\{\hat{Q}^S\} &= \gamma_5\delta\{\mathrm{M}_{\mathrm{oo}}^{-1}\,\mathrm{M}_{\mathrm{oe}}\,\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\}\\
&= -\gamma_5\,\mathrm{M}_{\mathrm{oo}}^{-1}\left[\,\mathrm{M}_{\mathrm{oe}}\,\mathrm{M}_{\mathrm{ee}}^{-1},\,\mathbf{1}\right]\begin{bmatrix}0 & \delta\{\mathrm{M}_{\mathrm{eo}}\}\\ \delta\{\mathrm{M}_{\mathrm{oe}}\} & 0\end{bmatrix}\begin{bmatrix}\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\\ \mathbf{1}\end{bmatrix}\\
&\quad + \gamma_5\,\mathrm{M}_{\mathrm{oo}}^{-1}\left[\,\mathrm{M}_{\mathrm{oe}}\,\mathrm{M}_{\mathrm{ee}}^{-1},\,\mathbf{1}\right]\begin{bmatrix}\delta\{\mathrm{M}_{\mathrm{ee}}\} & 0\\ 0 & \delta\{\mathrm{M}_{\mathrm{oo}}\}\end{bmatrix}\begin{bmatrix}\mathbf{1}\\ \mathrm{M}_{\mathrm{oo}}^{-1}\,\mathrm{M}_{\mathrm{oe}}\end{bmatrix}\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}} \qquad (5.19)\\
\delta\{\hat{Q}^{S\dagger}\} &= \gamma_5\delta\{\mathrm{M}_{\mathrm{oe}}\,\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\,\mathrm{M}_{\mathrm{oo}}^{-1}\}\\
&= -\gamma_5\left[\,\mathrm{M}_{\mathrm{oe}}\,\mathrm{M}_{\mathrm{ee}}^{-1},\,\mathbf{1}\right]\begin{bmatrix}0 & \delta\{\mathrm{M}_{\mathrm{eo}}\}\\ \delta\{\mathrm{M}_{\mathrm{oe}}\} & 0\end{bmatrix}\begin{bmatrix}\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\\ \mathbf{1}\end{bmatrix}\mathrm{M}_{\mathrm{oo}}^{-1}\\
&\quad + \gamma_5\,\mathrm{M}_{\mathrm{oe}}\,\mathrm{M}_{\mathrm{ee}}^{-1}\left[\,\mathbf{1},\,\mathrm{M}_{\mathrm{eo}}\,\mathrm{M}_{\mathrm{oo}}^{-1}\right]\begin{bmatrix}\delta\{\mathrm{M}_{\mathrm{ee}}\} & 0\\ 0 & \delta\{\mathrm{M}_{\mathrm{oo}}\}\end{bmatrix}\begin{bmatrix}\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\\ \mathbf{1}\end{bmatrix}\mathrm{M}_{\mathrm{oo}}^{-1}, \qquad (5.20)
\end{aligned}$$

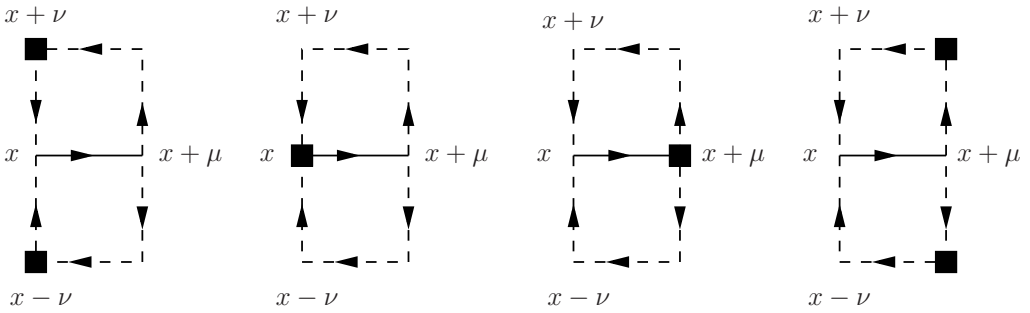

Figure 5.3. The contributing diagrams to $F_\mu^{\mathrm{sw}}(x)$.

allowing us to write (5.18) as

$$\delta\{S_b\} = -\left(X^\dagger\gamma_5 \begin{bmatrix} 0 & \delta\{\mathrm{M_{eo}}\} \\ \delta\{\mathrm{M_{oe}}\} & 0 \end{bmatrix} Y + X^\dagger\gamma_5 \begin{bmatrix} \delta\{\mathrm{M_{ee}}\} & 0 \\ 0 & \delta\{\mathrm{M_{oo}}\} \end{bmatrix} Z\right) + \mathrm{H.\,c.} \quad (5.21)$$

with

$$X = \begin{bmatrix} -\mathrm{M_{ee}^{-1}}\,\mathrm{M_{eo}} \\ \mathbf{1} \end{bmatrix} \mathrm{M_{oo}^{-1}} \left(\hat{Q}^S\hat{Q}^{S\dagger}\right)^{-1}\phi \quad (5.22)$$

$$Y = \begin{bmatrix} -\mathrm{M_{ee}^{-1}}\,\mathrm{M_{eo}} \\ \mathbf{1} \end{bmatrix} \hat{Q}^{S-1}\phi \quad (5.23)$$

$$Z = \begin{bmatrix} -\mathbf{1} \\ \mathrm{M_{oo}^{-1}}\,\mathrm{M_{oe}} \end{bmatrix} \mathrm{M_{ee}^{-1}}\,\mathrm{M_{eo}}\,\hat{Q}^{S-1}\phi. \quad (5.24)$$

The force F_μ^b splits into one contribution from the hopping terms and one from the Sheikholeslami-Wohlert terms. The hopping force for an even x is given by[1]

$$F_\mu^{\mathrm{hop}}(x) = \kappa\,\mathrm{Tr}_{\,\mathrm{Dirac}}\Big\{\gamma_5(1-\gamma_\mu)\Big[Y_o(x+\mu)\otimes X_e^\dagger(x) + X_o(x+\mu)\otimes Y_e^\dagger(x)\Big]\Big\}. \quad (5.25)$$

For the force F_μ^{sw} resulting from the Sheikholeslami-Wohlert terms we have to consider like for the determinant contribution the staples containing the link pointing form x to $x+\mu$ as they are shown in Fig. 5.3 and insert for the black boxes the matrix in color space [32]

$$(\blacksquare)_x = \mathrm{Tr}_{\,\mathrm{Dirac}}\left\{\gamma_5\sigma_{\mu\nu}Z(x)\otimes X(x)^\dagger + \mathrm{H.\,c.}\right\}. \quad (5.26)$$

The force F_μ^{sw} is then given by summing over all diagrams for $\mu \neq \nu$ and multiplying the result with $i\kappa c_{\mathrm{sw}}/8$.

[1] For x odd e is to be replaced by o and vice versa.

Hence the bosonic force is the sum of both parts

$$F_\mu^b(x) = \kappa \,\mathrm{Tr}_{\,\mathrm{Dirac}} \Big\{ \gamma_5 (1-\gamma_\mu) \Big[Y_o(x+\mu) \otimes X_e^\dagger(x) + X_o(x+\mu) \otimes Y_e^\dagger(x) \Big] \Big\} + \frac{i\kappa c_{\mathrm{sw}}}{8} \sum_{\mu \neq \nu} (\text{diagrams in Fig. 5.3}). \tag{5.27}$$

Comparing the three contributions to the total force we find that the computation of the bosonic force is most expensive, but the gauge force has the largest impact. One can account for those differences by improving the integration scheme as is discussed in the next section and demonstrated in Chapter 7.

5.4 Integrator

The integrator used to evolve the EoM must obey two conditions thus the HMC algorithm satisfies detailed balance. First it must be *reversible*, i.e. for a negative iteration step $-\delta\tau$ we must return (within the numerical precision) to an "earlier" state. Secondly the area in phase space must be preserved.

The second order *leapfrog* integration scheme is the simplest exhibiting these properties.[8] To integrate a trajectory of length τ, steps of size $\delta\tau = \tau/n_{\mathrm{step}}$ are performed. The integration starts (ends) by an initial (final) half step updating the conjugate momenta Π_μ, thus Π_μ and U_μ are always updated at times differing by $\delta\tau/2$ as the following scheme indicates

$$\begin{aligned}
&\Pi_\mu \to \Pi_\mu - \delta\{S_{\mathrm{eff}}\}/\delta\{U_\mu\} \cdot \delta\tau/2 && \text{initial half step} \\
&\quad \begin{bmatrix} U_\mu \to \exp\{\Pi_\mu \delta\tau\}\, U_\mu \\ \Pi_\mu \to \Pi_\mu - \delta\{S_{\mathrm{eff}}\}/\delta\{U_\mu\} \cdot \delta\tau \end{bmatrix} && \times (n_{\mathrm{step}} - 1) \\
&U_\mu \to \exp\{\Pi_\mu \delta\tau\}\, U_\mu && \text{final } U_\mu \text{ step} \\
&\Pi_\mu \to \Pi_\mu - \delta\{S_{\mathrm{eff}}\}/\delta\{U_\mu\} \cdot \delta\tau/2 && \text{final half step}
\end{aligned}$$

The error due to the numerical integration is $O(\delta\tau^2)$ for the two half steps and $O(\delta\tau^3)$ for the intermediate steps resulting in a total error for all n_{step} steps of the entire trajectory of $O(\tau\,\delta\tau^2)$.

The numerical error of the integration can be reduced by considering an improved integrator. Sexton and Weingarten proposed the following frequently used scheme [50]

$$
\begin{aligned}
&\Pi_\mu \to \Pi_\mu - \delta\{S_{\text{eff}}\}/\delta\{U_\mu\} \cdot \delta\tau/6 && \text{initial step}\\
&\quad\left[\begin{array}{l} U_\mu \to \exp\{\Pi_\mu \delta\tau/2\}\, U_\mu \\ \Pi_\mu \to \Pi_\mu - \delta\{S_{\text{eff}}\}/\delta\{U_\mu\} \cdot 2\delta\tau/3 \\ U_\mu \to \exp\{\Pi_\mu \delta\tau/2\}\, U_\mu \\ \Pi_\mu \to \Pi_\mu - \delta\{S_{\text{eff}}\}/\delta\{U_\mu\} \cdot \delta\tau/3 \end{array}\right] && \times (n_{\text{step}} - 1)\\
&U_\mu \to \exp\{\Pi_\mu \delta\tau/2\}\, U_\mu && \text{final steps}\\
&\Pi_\mu \to \Pi_\mu - \delta\{S_{\text{eff}}\}/\delta\{U_\mu\} \cdot 2\delta\tau/3\\
&U_\mu \to \exp\{\Pi_\mu \delta\tau/2\}\, U_\mu\\
&\Pi_\mu \to \Pi_\mu - \delta\{S_{\text{eff}}\}/\delta\{U_\mu\} \cdot \delta\tau/6
\end{aligned}
$$

Since the error is reduced a somewhat larger step size becomes possible. The Sexton-Weingarten integrator assigns different time scales to U_μ and Π_μ, which is also named multiple time scale integration (MTS or MTSI), see Section 7.1.2. Further improvements are proposed by the class of symplectic force-gradient integrator.[51]

Both presented integration schemes start (end) by performing an initial (final) step on Π_μ. By no means this is a constraint and updating first U_μ is valid, too. Aoki et al. even report that this version leads to a higher acceptance rate at slightly reduced costs.[33]

5.5 Multi-Pseudo-Fermion Fields

The most expensive part of the HMC update is the bosonic contribution to the force since at every integration step the matrix $(\hat{Q}\hat{Q}^\dagger)$ has to be inverted. For the numerical inversion of a positive-definite and symmetric matrix commonly a *conjugate gradient* (CG) is employed (cf. Appendix A.3). The costs of the inversion are related to the *condition number*[2] of the matrix to be inverted. Moreover, the related fermionic forces necessitate a small step-size $\delta\tau$.

Therefore, the idea of introducing several pseudo-fermions aims at splitting the bosonic force into several smaller parts which are cheaper to compute and reduce the statistical fluctuations in the total force.

[2]The condition number is defined as ratio of the largest over the smallest eigenvalue of a matrix.[29]

5.5.1 Hasenbusch-Trick

Martin Hasenbusch came up with the idea to reduce the absolute bosonic force making use of the identity $A = A' \cdot (A'^{-1}A)$. This allows to factorize the determinant into $\det\{A\} = \det\{A'\} \cdot \det\{A'^{-1}A\}$ and for each factor pseudo-fermion fields are introduced. In his first proposal a second, lower κ parameter is introduced allowing for two pseudo-fermions.[52] We consider this idea later in Chapter 6. Here we focus at a variant of the Hasenbusch-trick presented in [53], where a shift ρ is added to $\hat{Q}$. In a generalized form allowing for an arbitrary number of pseudo-fermions N_{PF}, the Hasenbusch-trick creates the sequence of Dirac-Wilson operators [54]

$$\begin{aligned} W_1 &= \hat{Q} + \rho_1 \\ W_i &= (\hat{Q} + \rho_{i-1})^{-1}(\hat{Q} + \rho_i) \\ W_N &= (\hat{Q} + \rho_{N-1})^{-1}\hat{Q}. \end{aligned} \tag{5.28}$$

In the end the total force of all pseudo-fermion fields proves to be smaller then the original force which leads to an improved algorithm. Introducing more pseudo-fermion fields also introduces more noise which decreases the acceptance. Thus, in principle, an optimal number of pseudo-fermion fields has to be found. Splitting the action into pieces allows furthermore to give each contribution its own time scale (cf. Section 7.1.2).

Looking at eq. (5.28) you notice only the first pseudo-fermion field is governed by a simple expression while all others have Dirac operators formed by a ratio of shifted fermion matrices. Hence we expect the computational cost to be cheap for the first and expensive for the remaining operators. Therefore the gain of introducing more than two pseudo-fermion fields is expected to be small. For the sake of simplicity, we restrict ourselves to $N_{PF} = 2$, where only one additional parameter ρ is introduced. Moreover, we restrict ourselves to the symmetrically preconditioned Dirac-Wilson operator $\hat{Q}^S$ dropping subsequently the label "S". Starting from the determinant of $\hat{Q}^\dagger\hat{Q}$ we derive the actions of the two pseudo-fermions and the corresponding operators

$$\begin{aligned} \det\{\hat{Q}^\dagger\hat{Q}\} &= \det\{W_1 W_1^\dagger\} \cdot \det\{[W_1^{-1}\hat{Q}][W_1^{-1}\hat{Q}]^\dagger\} \\ &\propto \int \mathcal{D}\phi_1^\dagger \, \mathcal{D}\phi_1 \, \mathcal{D}\phi_2^\dagger \, \mathcal{D}\phi_2 \; \exp\left\{ -\sum_{i=1}^{2} S_{F_i} \right\} \end{aligned} \tag{5.29}$$

$$S_{F_1} = \phi_1^\dagger (W_1 W_1^\dagger)^{-1} \phi_1 \tag{5.30}$$

$$S_{F_2} = \phi_2^\dagger \left([W_1^{-1}\hat{Q}][W_1^{-1}\hat{Q}]^\dagger \right)^{-1} \phi_2 = \phi_2^\dagger (W_2 W_2^\dagger)^{-1} \phi_2. \tag{5.31}$$

Since the symmetric preconditioned operator $\hat{Q}$ is not Hermitian some equations are simplified by multiplying the (real) shift ρ by $\mathrm{M}_{\mathrm{oo}}^{-1}$. Hence we find

$$W_1 = \hat{Q} - i\rho\,\mathrm{M}_{\mathrm{oo}}^{-1} \tag{5.32}$$

$$W_2 = W_1^{-1}\hat{Q} = (\hat{Q} - i\rho\,\mathrm{M}_{\mathrm{oo}}^{-1})^{-1}\hat{Q}, \tag{5.33}$$

and create the two pseudo-fermions by applying W_1 or W_2, respectively, to a Gaussian vector η

$$\phi_1 = W_1\eta = \left(\hat{Q} - i\rho\,\mathrm{M}_{\mathrm{oo}}^{-1}\right)\eta \tag{5.34}$$

$$\phi_2 = W_2\eta = (\hat{Q} - i\rho\,\mathrm{M}_{\mathrm{oo}}^{-1})^{-1}\hat{Q}\,\eta. \tag{5.35}$$

In case of the second pseudo-fermion we are forced to compute the inversion of W_1. Hence the second pseudo-fermion becomes "expensive" compared to the first one.

Repeating the computation to get the fermionic forces we find by varying S_{F_1} the three vectors X_1, Y_1 and Z_1 entering in equation (5.21)

$$X_1 = \begin{bmatrix} -\mathrm{M}_{\mathrm{ee}}^{-1}\mathrm{M}_{\mathrm{eo}} \\ \mathbb{1} \end{bmatrix} \mathrm{M}_{\mathrm{oo}}^{-1} \left(W_1 W_1^\dagger\right)^{-1} \phi_1 \tag{5.36}$$

$$Y_1 = \begin{bmatrix} -\mathrm{M}_{\mathrm{ee}}^{-1}\mathrm{M}_{\mathrm{eo}} \\ \mathbb{1} \end{bmatrix} W_1^{-1}\phi_1 \tag{5.37}$$

$$Z_1 = \left(\begin{bmatrix} -\mathbf{1} \\ \mathrm{M}_{\mathrm{oo}}^{-1}\mathrm{M}_{\mathrm{oe}} \end{bmatrix} \mathrm{M}_{\mathrm{ee}}^{-1}\mathrm{M}_{\mathrm{eo}} + \begin{bmatrix} 0 \\ i\rho\gamma_5\,\mathrm{M}_{\mathrm{oo}}^{-1} \end{bmatrix}\right) W_1^{-1}\phi_1. \tag{5.38}$$

Re-writing the action of the second pseudo-fermion, S_{F_2}, we notice that it is the *asymmetric* even-odd preconditioned operator (3.8) governing this pseudo-fermion[3]

$$S_{F_2} = \phi_2^\dagger (W_2 W_2^\dagger)^{-1}\phi_2 = \phi_2^\dagger \left(\mathbb{1} + \rho^2 Q^{A-2}\right)\phi_2. \tag{5.39}$$

Computing the vectors entering the in the force computation we find

$$X_2 = \begin{bmatrix} -\mathrm{M}_{\mathrm{ee}}^{-1}\mathrm{M}_{\mathrm{eo}} \\ \mathbb{1} \end{bmatrix} \left(\hat{Q}^A \hat{Q}^{A\dagger}\right)^{-1} \rho\phi_2 = \begin{bmatrix} -\mathrm{M}_{\mathrm{ee}}^{-1}\mathrm{M}_{\mathrm{eo}} \\ \mathbb{1} \end{bmatrix} \mathrm{M}_{\mathrm{oo}}^{-1} \left(\hat{Q}\hat{Q}^\dagger\right)^{-1} \mathrm{M}_{\mathrm{oo}}^{-1}\,\rho\phi_2 \tag{5.40}$$

$$Y_2 = Z_2 = \begin{bmatrix} -\mathrm{M}_{\mathrm{ee}}^{-1}\mathrm{M}_{\mathrm{eo}} \\ \mathbb{1} \end{bmatrix} \hat{Q}^{A-1}\rho\phi_2 = \begin{bmatrix} -\mathrm{M}_{\mathrm{ee}}^{-1}\mathrm{M}_{\mathrm{eo}} \\ \mathbb{1} \end{bmatrix} \hat{Q}^{-1}\,\mathrm{M}_{\mathrm{oo}}^{-1}\,\rho\phi_2. \tag{5.41}$$

5.5.2 n^{th} Root-Trick

Another way to factorize the fermion determinant is to take the n^{th} power of the n^{th} root of the matrix $MM^\dagger$ discussed by Tony Kennedy in detail in [27] (see also [55])

$$\det\{MM^\dagger\} = \left[\det\{MM^\dagger\}^{1/n}\right]^n. \tag{5.42}$$

[3] Usually, one absorbs the factor ρ in the ϕ_2 fields.

He motivates taking the n^{th} root by considering the condition number because $\text{cond}(\{MM^\dagger\}^{1/n}) = \sqrt[n]{\text{cond}(MM^\dagger)}$ which alleviates the inversion of $MM^\dagger$ significantly and also lowers the appearing forces. The latter becomes apparent if one assumes that the force from the pseudo-fermions acting on the gauge field is inversely proportional to the smallest eigenvalue of $MM^\dagger$ (at least for sufficiently small fermion masses) and that the largest eigenvalue of $MM^\dagger$ is essentially fixed. Hence the total force from all n pseudo-fermions is proportional to $n \cdot \text{cond}(\{MM^\dagger\}^{1/n}) < \text{cond}(MM^\dagger)$, where also is assumed that all pseudo-fermions contribute equally. Due to the n^{th} root the force is reduced by the factor $c = n \cdot \text{cond}(MM^\dagger)^{(1-n)/n}$ allowing for an step size increased by c^{-1} which leads to a cost reduction by the factor c. Therefrom follows the optimal number of pseudo-fermions

$$n_{\text{opt}} \sim \ln\{\text{cond}(MM^\dagger)\}. \tag{5.43}$$

Advantageous of this factorization is that "symmetric" pseudo-fermion fields are created. All Dirac operators have the same condition number. The disadvantage is that the approximation of $M^{-1/n}$ is required. Consequently, it can not be exploited as easy as the Hasenbusch-trick. An application is discussed within the rational HMC presented as variant of the basic HMC algorithm later on.

5.6 Variants

Since the publication of the basic HMC algorithm more than 20 years ago many improvements and variations have been proposed. Here we discuss shortly some but certainly not all those of greater relevance.

5.6.1 PHMC

Inspired by the multi-boson algorithm [56] and a suggestion presented in [57], Frezzotti and Jansen proposed to express the squared, inverse Hermitian Dirac-Wilson Operator by Chebyshev polynomials. Thereby they deviate from importance sampling and account for that by reweighting.[58–60]

The inverse of $(\hat{Q}^A)^2$ is approximated by the Chebyshev polynomial given here in its product representation

$$P_{n,\epsilon}((\hat{Q}^A)^2) = c_N \prod_{k=1}^{n} \left[\left(\hat{Q}^A - \sqrt{z_k^*}\right)\left(\hat{Q}^A - \sqrt{z_k}\right)\right], \tag{5.44}$$

where z_k are the complex roots of the polynomial arising in complex pairs

$$z_k = \frac{1+\epsilon}{2} - \frac{1+\epsilon}{2}\cos\left(\frac{2\pi k}{n+1}\right) - i\sqrt{\epsilon}\sin\left(\frac{2\pi k}{n+1}\right), \tag{5.45}$$

and c_N is an explicitly calculable coefficient.[61] The polynomial approximates the spectrum of the normalized[4] operator $(\hat{Q}^A)^2$ in the interval $\lambda \in [\epsilon, 1]$ with a relative fit error bounded from above by

$$\delta \equiv 2\left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)^{n+1}. \tag{5.46}$$

To keep rounding errors small, one is forced to reorder the roots $\sqrt{z_k}$ of the root factorized polynomial (5.44) in a suitable way preserving the relation $\sqrt{z_{2n+1-k}} = \sqrt{z_k^*}$.[62]

The general outline of the update is used as basis for the implementation of the new variant approximating $\hat{M}$ (called NPHMC) and is in detail described in Chapter 6. One further difficulty of this variant is that the computation of the variation of the polynomial action becomes numerically instable due to eigenvalues "very close to 1". This can be avoided by adjusting the normalization such that the eigenvalue lie within $\lambda \in [\epsilon, 0.9]$ or by employing an improved recurrence relation (Clenshaw recursion) as described in [63]. The reweighting factor compensating for a deviation from importance sampling is computed by

$$C = \exp\left\{\eta^\dagger\left(1 - \left[(Q^A)^2 P_{n,\epsilon}((\hat{Q}^A)^2)\right]^{-1}\right)\eta\right\}, \tag{5.47}$$

where η is random Gaussian vector.

A quite similar variant presented by Aoki et al. is also based on the root factorization but approximates the inverse, non-Hermitian, symmetrically preconditioned operator $\hat{M}^{S-1}$ by P_n^S and computes the recursions by a Horner scheme.[33, 64, 65] Furthermore, they replace the reweighting factor by a second accept-reject step as suggested in [66, 67] and accept a new configuration according to

$$P_{\text{corr}}[U \to U'] = \min\{1, \exp\{-R\}\} \tag{5.48}$$

with $R = |(P_n^S \hat{M}^S[U'])^{-1} P_n^S \hat{M}^S[U]\eta| - |\eta|$ and η a Gaussian vector of unit variance and zero mean.

[4] Normalization is achieved introducing an appropriate factor in eq. (3.8).

Multi-Step Multi-Boson Algorithm

Montvay and Scholz extended the PHMC by applying mass-preconditioning [55] in a way especially suitable for the PHMC algorithm.[68, 69] Introducing the concept of a multi-step algorithm the inverting polynomial is obtained by the recursive description

$$P_i \simeq [(x+\rho_i)^\alpha P_1(x)\cdots P_{i-1}(x)]^{-1}\,; \quad i=1,2,\ldots,k, \tag{5.49}$$

where $\alpha = 1/k$ and the ρ_i are positive with $\rho_k = 0$. Hence during the first steps a shifted x is approximated which requires only a lower degree polynomial if $\rho_i/\rho_{i-1} \lesssim 1$ and the acceptance remains sufficient. Similar to MTS integration schemes the "easy" part, here the cruder approximation may be done more often and only at the end the "expensive" computation (the inversion of x with the highest degree polynomial) has to be performed.

5.6.2 RHMC

Replacing the polynomial by a *rational* approximation Clark and Kennedy formulated the rational Hybrid Monte Carlo algorithm (RHMC) [27, 70, 71], where the fermion determinant is rewritten as

$$\begin{aligned}\det\{(M^\dagger M)^\alpha\} &= \int \mathcal{D}\phi^\dagger\,\mathcal{D}\phi\ \exp\{-\phi^\dagger (M^\dagger M)^{-\alpha}\phi\} \\ &\approx \int \mathcal{D}\phi^\dagger\,\mathcal{D}\phi\ \exp\{-\phi^\dagger r^2(M^\dagger M)\phi\}, \end{aligned}\tag{5.50}$$

$$\text{with} \qquad r(x) = \sum_{k=1}^{n} \frac{\alpha_k}{x+\beta_k} \approx x^{-\alpha/2}. \tag{5.51}$$

The rational kernel $r(x)$ is generated by the Remez algorithm leading to a very precise approximation of $x^{-\alpha/2}$, which has a much better convergence than a polynomial approximation.[5][27] Thus one omits a reweighting step and covers the entire spectrum of $M^\dagger M$ right away. $r(x)$ is written as partial fractions (5.51) which are cost efficiently evaluated using a multi-shift solver[72]. Important facts of the rational approximation are that the roots and poles are in general real, for $|\alpha| < 1$ the poles are even positive and the α_k have the same sign. This leads to a numerically stable algorithm. The reason why this happens to be the case is still not understood.[27]

To make use of the rational approximation in the HMC algorithm one has to adjust the generation of the pseudo-fermions (5.17) by employing $r(M^\dagger M)$,

[5] Also the rational approximation is more expensive than the polynomial approximation for the same degree n, in the end one gains if a much lower degree for $r(x)$ is sufficient to yield the same precision.

$\phi = r(M^\dagger M)^{-1}\eta$. Moreover, a second, different rational approximation is used for evaluating the force

$$\bar{r} \approx (M^\dagger M)^{-\alpha} \approx r^2(M^\dagger M). \tag{5.52}$$

That way a double inversion is avoided and the pseudo-fermion force is given by a sum of HMC like terms

$$\delta\{S_{PF}\} = -\sum_{i=1}^{\overline{m}} \bar{\alpha}_i \phi^\dagger (M^\dagger M + \bar{\beta}_i)^{-1} \delta\{M^\dagger M\} (M^\dagger M + \bar{\beta}_i)^{-1} \phi. \tag{5.53}$$

In addition, the RHMC allows for the symmetric splitting of the determinant to introduce several pseudo-fermions (see Section 5.5.2) as applied in [73, 74]

$$\begin{aligned} \det\{M^\dagger M\} &= \left[\det\{(M^\dagger M)^{1/n}\}\right]^n \\ &\propto \int \prod_{j=1}^{n} \mathcal{D}\phi_j^\dagger \mathcal{D}\phi_j \ \exp\{-\phi_j^\dagger (M^\dagger M)^{-1/n} \phi_j\}. \end{aligned} \tag{5.54}$$

The n^{th} root-trick designs all fermions to have the same masses and hence forces of similar magnitude. Therefore, the use of an MTS integration scheme (on the pseudo-fermion fields) is not suitable. In [71] a first comparison between the similar n^{th} root fermions and Hasenbusch-fermions of different masses on multiple time slices is reported but no significant differences are seen. It is claimed that due to the higher order integrator used for n^{th} root fermions a superior volume scaling is expected.

5.6.3 DD-HMC

A different approach is presented by Lüscher which now goes under the name DD-HMC indicating that *domain-decomposition* methods are combined with the conventional HMC. In a series of publications [75–77] he discusses possibilities to apply the Schwarz-alternating procedure to lattice QCD algorithms and he finally focuses at the domain-decomposition method as preconditioner to the HMC algorithm. Similar to even-odd preconditioning the lattice is divided into non-overlapping rectangular blocks Λ differentiated like on a chessboard by "black" and "white". Naming the union of all black blocks Ω and the union of white blocks Ω^* the Dirac-Wilson D operator is written as

$$D = \begin{bmatrix} D_\Omega & D_{\partial\Omega} \\ D_{\partial\Omega^*} & D_{\Omega^*} \end{bmatrix}, \tag{5.55}$$

where D_Ω is the Dirac-Wilson operator on Ω with Dirichlet boundary conditions and $\mathrm{D}_{\partial\Omega}$ is the sum of all hopping terms connecting the boundary $\partial\Omega$

of Ω to the boundary $\partial\Omega^*$ of Ω^*. Defining the pseudo-fermion fields on the entire lattice we extend the operators by padding zeros

$$D = D_\Omega + D_{\Omega^*} + D_{\partial\Omega} + D_{\partial\Omega^*} \tag{5.56}$$

and we have as operator on all blocks Λ

$$D_\Omega + D_{\Omega^*} = \sum_{\text{all}\,\Lambda} D_\Lambda. \tag{5.57}$$

The block form (5.55) of the Dirac-Wilson operator allows us to factorize its determinant according to

$$\det\{D\} = \det D_\Omega \cdot \det D_{\Omega^*} \cdot \det\left\{1 - D_\Omega^{-1} D_{\partial\Omega} D_{\Omega^*}^{-1} D_{\partial\Omega^*}\right\}. \tag{5.58}$$

Combining the Schwarz-preconditioning with the standard even-odd preconditioning we find

$$\det D_\Omega \cdot \det D_{\Omega^*} = \prod_{\text{all}\,\Lambda} \det \hat{D}_\Lambda, \tag{5.59}$$

with $\hat{D}_\Lambda$ the even-odd preconditioned Dirac-Wilson operator (without $O(a)$ improvement). If we now split the determinant (5.58) into two factors we can introduce for each a pseudo-fermion field and enter the HMC algorithm

$$D_1 = \sum_{\text{all}\,\Lambda} \hat{D}_\Lambda \quad \text{and} \quad D_2 = 1 - P_{\partial\Omega^*} D_\Omega^{-1} D_{\partial\Omega} D_{\Omega^*}^{-1} D_{\partial\Omega^*}. \tag{5.60}$$

For the second factor it is important to note that the curly bracket in (5.58) acts non-trivially only on the components of the fermion fields residing in $\partial\Omega^*$ and by introducing an orthonormal projector ($P_{\partial\Omega^*}$) it can be restricted to that subspace.

Furthermore we note that the Dirac-Wilson operator couples only nearest neighbors and hence the equations on the black and white blocks are completely decoupled. A similar decoupling can be achieved for the preconditioned HMC if the molecular dynamics evolution is restricted to the *active links*. Active links are defined to start and end within one block and have at most one endpoint on the interior boundary of that block. Now the active links of different blocks are only coupled by D_2 which favors an integration scheme evolving most of the time these blocks in parallel and rarely integrating the block-interacting part D_2. This gets supported by the fact that the force resulting from D_2 is small compared to the one from D_1. Computing the force from D_2 is also the most expensive part since the full Dirac operator has to be inverted

$$D_2^{-1} = 1 - P_{\partial\Omega^*} D^{-1} D_{\partial\Omega}. \tag{5.61}$$

Here, a further speedup can be achieved by using low-mode deflation techniques.[78]

Chapter 6

Non-Hermitian Polynomial Hybrid Monte Carlo

In the preceding sections the importance of even-odd-preconditioning and the advantages of the symmetric version are shown. Therefore we start now from the partition function (3.14) and modify it for the polynomial approximation. Let us first focus on the determinant of the even-odd preconditioned Dirac-Wilson operator and relate them to the polynomial approximation of its inverse

$$\begin{aligned}\det\left\{\hat{M}\hat{M}^\dagger\right\} = \det\left\{\left[\hat{M}P_n(\hat{M})\right]\left[\hat{M}P_n(\hat{M})\right]^\dagger\right\} \\ \cdot\left[\det\left\{P_n(\hat{M})P_n(\hat{M})^\dagger\right\}\right]^{-1}.\end{aligned} \tag{6.1}$$

The first term gives rise to the *correction factor* C [59], while the second factor enters the effective action. We find for the partition function

$$\mathcal{Z} = \int\mathcal{D}U\,\mathcal{D}\phi^\dagger\,\mathcal{D}\phi\,\mathcal{D}\eta^\dagger\,\mathcal{D}\eta\;C\;e^{-S_G-S_b-S_\eta-S_{\det}} \tag{6.2}$$

$$S_b = \phi^\dagger P_n^\dagger(\hat{M})P_n(\hat{M})\phi$$

$$S_\eta = \eta^\dagger\eta$$

$$C = \exp\{\eta^\dagger[\mathbb{1} - (P_n^\dagger(\hat{M})\hat{M}^\dagger\hat{M}P_n(\hat{M}))^{-1}]\eta\}. \tag{6.3}$$

As before the determinant is replaced by a bosonic Gaussian integral $\det A \propto \int \mathrm{d}\chi^\dagger\mathrm{d}\chi\exp\{-\chi^\dagger A^{-1}\chi\}$. The correction factor C may be neglected in the beginning, assuming a high degree polynomial approximating $\hat{M}^{-1}$ well, thus $C = 1$. The determinant contribution $S_{\det}$ depends on the choice of preconditioning and is for the symmetric version given by (3.17)

$$S_{\det} = -2[\ln\det\{\mathrm{M_{ee}}\} + \ln\det\{\mathrm{M_{oo}}\}]. \tag{6.4}$$

This contribution will as before be computed exactly (cf. Section 5.3, eq. (5.12) f.). η labels again a Gaussian random vector and hence S_η is easily computed and the contribution of the gauge fields S_G is unchanged in comparison with the standard HMC as described in the previous section. Due to the polynomial approximation the bosonic contribution S_b changes and in the following sections we derive the various parts required in the update. Nevertheless the general structure remains unchanged and the main evolution is performed on the odd sites only like in case of the standard HMC.

Using the (symmetric) even-odd preconditioned operator the polynomial approximation favors strongly to incorporate the shift of the spectrum along the real axis. Hence in the following we use

$$\hat{K}^S = \widetilde{K} - \hat{\delta}\,\mathbb{1} \tag{6.5}$$

$$\hat{M}^S = \mathbb{1} - \hat{K}^S = (1+\hat{\delta})\,\mathbb{1} - \widetilde{K} = d\,\mathbb{1} - \widetilde{K}, \tag{6.6}$$

as well as the more general recurrence relations for the corresponding polynomials

$$\begin{aligned} R_{n+1}(\hat{M}) &= a_n \widetilde{K} R_n(\hat{M}) + (1 - d a_n) R_{n-1}(\hat{M}) \\ &\qquad \text{with} \quad R_1 = \widetilde{K}/d \quad \text{and} \quad R_0 = \mathbb{1}; \end{aligned} \tag{6.7}$$

$$\begin{aligned} P_n(\hat{M}) &= a_n(\mathbb{1} + \widetilde{K} P_{n-1}) + (1 - d a_n) P_{n-2}(\hat{M}) \\ &\qquad \text{with} \quad P_1 = a_1(\mathbb{1} + \widetilde{K}/d) \quad \text{and} \quad P_0 = \mathbb{1}/d; \end{aligned} \tag{6.8}$$

$$a_n = \left(d - \hat{e}^2 a_{n-1}/4\right)^{-1} \quad \text{with} \quad a_1 = \left(d - \hat{e}^2/(2d)\right)^{-1}. \tag{6.9}$$

Consequently, our approximation has three input parameters:
n the degree of the polynomials, $\hat{e}$ the eccentricity of the ellipse determining the region to be approximated and $d = 1 + \hat{\delta}$ the shift of this ellipse along the real axis.

6.1 Creating the Bosonic Fields

The bosonic fields ϕ_o are generated from a Gaussian vector (η_o) and have to be multiplied by the inverse of the inverting polynomial to "feel" the appropriate sampling. Since P_n^{-1} is not easily available we insert $\mathbb{1} = \hat{M}^{-1}\hat{M}$ and obtain

$$\phi_o = P_n^{-1}\eta_o = (\hat{M}P_n)^{-1}\hat{M}\eta_o = (\mathbb{1} - R_{n+1})^{-1}\hat{M}\eta_o. \tag{6.10}$$

In the last step we are forced to invert $(\mathbb{1} - R_{n+1})$ which is a well conditioned matrix. Thus the inversion should be easy and need only a couple of iterations. We employ a *conjugate gradient* (CG) and invert $(\mathbb{1} - R_{n+1})(\mathbb{1} - R_{n+1})^\dagger$

because $(\mathbb{1} - R_{n+1})$ is not positive and symmetric as required by the CG. ϕ_o is therefore computed by

$$\phi_o = \left[(\mathbb{1} - R_{n+1})^\dagger (\mathbb{1} - R_{n+1})\right]^{-1} (\mathbb{1} - R_{n+1})^\dagger M \eta_o. \tag{6.11}$$

Applying R_{n+1} to a vector we can make use of the recurrence relation (6.7) which becomes a recursion of the vector fields on odd sites (χ) only. Using in addition (6.9) and starting with the computation of the two initial vectors

$$\chi_0 = \Phi_o \quad \text{and} \quad \chi_1 = \left(\widetilde{K}/d\right) \Phi_o,$$

we yield successively χ_{j+1} via

$$\chi_{j+1} = a_j \widetilde{K} \chi_j + (1 - d a_j) \chi_{j-1}. \tag{6.12}$$

6.2 Bosonic Forces

The bosonic forces arise by calculating the variation of the bosonic contribution to the action.[32, 34] The bosonic part of the effective action reads

$$S_b = \sum_x \phi_o^\dagger P_n^\dagger P_n \phi_o, \tag{6.13}$$

where the bosonic fields ϕ_o are generated according to the description given before. Computing the variation of (6.13) the variation of the polynomial has to be computed

$$\delta\{S_b\} = \phi_o^\dagger \left(P_n^\dagger \delta\{P_n\} + \delta\{P_n^\dagger\} P_n\right) \phi_o \tag{6.14}$$

First, we turn the matrix recurrence relation (6.8) into a recurrence relation of the χ-fields, which are again only defined on the odd sites. Hence the application of P_n to a vector ϕ_o is computed by

$$\begin{aligned}
&\chi_0 = \phi_o / d \\
&\chi_1 = a_1 \left(\mathbb{1} + \left(\widetilde{K}/d\right)\right) \phi_o \\
&\chi_j = a_j \phi_o + a_j \widetilde{K} \chi_{j-1} + (1 - d a_j) \chi_{j-2} \qquad \text{for } j = 2, \ldots, n \\
&\psi_o = \chi_n = P_n \phi_o,
\end{aligned} \tag{6.15}$$

with the coefficients a_n given by eq. (6.9). Computing now the scalar product $\chi_n^\dagger \chi_n$ we get moreover the bosonic contribution to the action.

Next we focus on calculating the variation of $\delta\{\psi_o\} = \delta\{\chi_n\}$

$$\delta\{\chi_n\} = a_n\delta\{\widetilde{K}\}\chi_{n-1} + a_n\widetilde{K}\delta\{\chi_{n-1}\} + (1 - da_n)\delta\{\chi_{n-2}\} \tag{6.16}$$

which starts with

$$\delta\{\chi_1\} = (a_1/d)\,\delta\{\widetilde{K}\}\phi_0 \quad \text{and} \quad \delta\{\chi_0\} = 0.$$

Looking at the recursion relation (6.16) one observes that the variation $\delta\{\widetilde{K}\}$ propagates through each iteration step. Reorganizing the expressions we obtain a sum, where only the l^{th} occurrence of $\widetilde{K}$ is varied

$$\delta\{\chi_n\} = \sum_{l=1}^{n} Q_{n-l}a_l\delta\{\widetilde{K}\}\chi_{l-1}. \tag{6.17}$$

and we additionally introduce the polynomials Q_j on the left of $\delta\{\widetilde{K}\}$ which obey the recursion

$$\begin{aligned} Q_j &= a_{n-j+1}\widetilde{K}Q_{j-1} + (1 - da_{n-j+2})Q_{j-2} \\ &\text{with} \quad Q_1 = a_n\widetilde{K} \quad \text{and} \quad Q_0 = \mathbb{1}\,. \end{aligned} \tag{6.18}$$

Finally, we obtain the variation of the action (6.14) by multiplying (6.17) with $\psi_o^\dagger = \chi_n^\dagger$ which allows us to apply Q_j on $\chi_n^\dagger$ and express (6.18) as vector recurrence relation defining $\xi_{n+j}^\dagger$

$$\begin{aligned} \xi_n^\dagger &= \chi_n^\dagger \\ \xi_{n+1}^\dagger &= \chi_n^\dagger\widetilde{K}a_n \\ \xi_{n+j}^\dagger &= \xi_{n+j-1}^\dagger\widetilde{K}a_{n-j+1} + \xi_{n+j-2}^\dagger(1 - da_{n-j+2}). \end{aligned} \tag{6.19}$$

Hence the variation of the bosonic contribution is

$$\begin{aligned} \delta\{S_b\} &= \left[\psi_o^\dagger\delta\{\psi_o\} + \delta\{\psi_o^\dagger\}\psi_o\right] \\ &= \sum_{l=1}^{n}\left[\xi_{2n-l}^\dagger a_l\delta\{\widetilde{K}\}\chi_{l-1} + \chi_{l-1}^\dagger\delta\{\widetilde{K}^\dagger\}a_l\,\xi_{2n-l}\right], \end{aligned} \tag{6.20}$$

which according to (5.9) is expressed as an infinitesimal change of the gauge link $\delta\{U_\mu(x)\}$.

Due to the sum in the last equation the computation of $\delta\{S_b\}$ becomes more cumbersome compared to the standard HMC since n contributions must be computed and added. For *symmetric* even-odd preconditioning $\delta\{\widetilde{K}\}$ and $\delta\{\widetilde{K}^\dagger\}$ are given by

$$\delta\{\widetilde{K}\} = -\gamma_5\delta\{\hat{Q}^S\} \quad \text{and} \quad \delta\{\widetilde{K}^\dagger\} = -\gamma_5\delta\{\hat{Q}^{S\dagger}\} \tag{6.21}$$

with the variations of $\hat{Q}^S$ and $\hat{Q}^{S\dagger}$ stated in (5.19) and (5.20), respectively.

Thus the sum in (6.20) is computed by generating for each l the three vectors

$$\begin{aligned} X^{\text{hop}} &= \begin{bmatrix} -\,\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\,\chi_{l-1} \\ \chi_{l-1} \end{bmatrix}; \qquad X^{\text{clover}} = \begin{bmatrix} -\,\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\,\chi_{l-1} \\ \mathrm{M}_{\mathrm{oo}}^{-1}\,\mathrm{M}_{\mathrm{oe}}\,\mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\,\chi_{l-1} \end{bmatrix}; \\ \Xi &= \begin{bmatrix} \mathrm{M}_{\mathrm{ee}}^{-1}\,\mathrm{M}_{\mathrm{eo}}\,a_l\,\mathrm{M}_{\mathrm{oo}}^{-1}\,\gamma_5\xi_{2n-l} \\ -a_l\,\mathrm{M}_{\mathrm{oo}}^{-1}\,\gamma_5\xi_{2n-l} \end{bmatrix} \end{aligned} \tag{6.22}$$

and computing each time the partial contribution

$$\begin{aligned} \delta\{S_b\}_l = \Xi^\dagger\gamma_5 \begin{bmatrix} 0 & \delta\{\mathrm{M}_{\mathrm{eo}}\} \\ \delta\{\mathrm{M}_{\mathrm{oe}}\} & 0 \end{bmatrix} X^{\text{hop}} + \Xi^\dagger\gamma_5 \begin{bmatrix} \delta\{\mathrm{M}_{\mathrm{ee}}\} & 0 \\ 0 & \delta\{\mathrm{M}_{\mathrm{oo}}\} \end{bmatrix} X^{\text{clover}} \\ + \text{H. c.} \end{aligned} \tag{6.23}$$

which when added result in $\delta\{S_b\}$. The multiplications performed in (6.23) are carried out as direct product in color space and moreover the trace in Dirac space is taken. The contribution given by the second summand and its Hermitian conjugate refer to the "black-box" insertions as introduced by Jansen and Liu in [32] (cf. Chapter 5).

The simplest strategy to implement the force computation is to first create and store all n χ-vectors and then start to calculate the contributions to $\delta\{S_b\}$ by iterating down from $l = n$ to 1. Obviously the drawback of this strategy is the large amount memory needed. If memory becomes a limiting factor Jansen and Frezzotti already proposed the following strategy (cf. [59] for a detailed description): Instead of storing all n χ-vectors only each, e.g., 10^{th} and 11^{th} vector are stored[1] (cf. Fig. 6.1 a) and subsequently we compute the direct products in a block of ten computations. First we re-compute from each stored pair the successive eight vectors (Fig. 6.1 b). By applying the same recurrence relation (6.15), we avoid any discrepancies due to different rounding. Now we can compute for this block of ten χ vectors the direct products with the ξ vectors which are generated on the fly by the recursion (6.19). Obviously, we are trading that way less memory requirement for in increase in computational costs by a factor two.

6.3 Correction Factor

At the end of a Monte Carlo trajectory the accept-reject step is performed to render the update of the algorithm exact. Nevertheless, we are not finished, since the correction factor C, eq. (6.3), has to be computed and the observables to be "reweighted".

[1] Unlike the original proposal we are forced due the two-step recursion to store two vectors for re-computation.

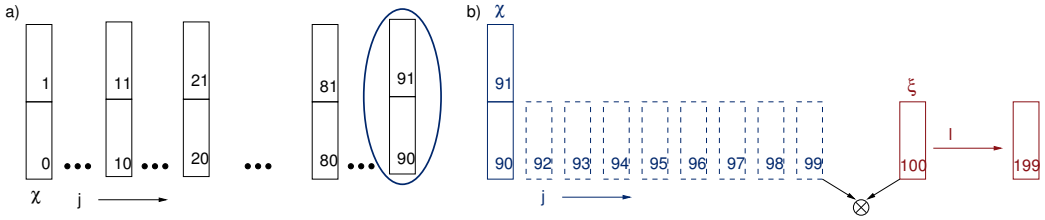

Figure 6.1. Computation with less memory consumption for polynomials of degree $n = 100$. a) Storing at first only each 10^{th} and 11^{th} χ vector, b) re-computing eight vectors before calculating the direct products. The stacking of the vectors indicates that for efficient memory usage the 11^{th} vectors are stored on the unused even components of the corresponding 10^{th} vector.

The correction factor is estimated according to

$$C = \langle \widehat{C} \rangle_\eta, \tag{6.24}$$

where $\widehat{C}$ is estimated using Gaussian numbers η

$$\widehat{C} = \exp\{\eta^\dagger [\mathbb{1} - ((\hat{M}P_n)^\dagger (\hat{M}P_n))^{-1}]\eta\}. \tag{6.25}$$

Again we express $\hat{M}P_n$ by $(\mathbb{1} - R_{n+1})$ and use a CG for the required inversion. We compute

$$\widehat{C} = \exp\{\eta^\dagger [\mathbb{1} - ((\mathbb{1} - R_{n+1})^\dagger (\mathbb{1} - R_{n+1}))^{-1}]\eta\} \tag{6.26}$$

by first creating

$$\zeta = \left[(\mathbb{1} - R_{n+1})^\dagger (\mathbb{1} - R_{n+1})\right]^{-1} (\mathbb{1} - R_{n+1})^\dagger \eta \tag{6.27}$$

and thus obtain $\widehat{C}$ by

$$\begin{aligned} \widehat{C} &= \exp\{\zeta^\dagger [(\mathbb{1} - R_{n+1}^\dagger)(\mathbb{1} - R_{n+1}) - \mathbb{1}]\zeta\} \\ &= \exp\{\zeta^\dagger [-R_{n+1}^\dagger - R_{n+1} + R_{n+1}^\dagger R_{n+1}]\zeta\}, \end{aligned} \tag{6.28}$$

The computation is analogously performed like in the heatbath, i.e. the recursion (6.12) is used to apply R_{n+1} to a vector. For a polynomial of high degree n approximating the inverse well, the correction factor C has the limit 1 but may deviate clearly from 1 for a lower degree polynomial. More relevant for the quality of the approximation than the value itself is the variance

of the correction factor giving us the width of its distribution. Therefore we consider in addition the normalized quantity

$$\varsigma_C = \frac{\sqrt{\langle C^2\rangle - \langle C\rangle^2}}{\langle C\rangle}, \tag{6.29}$$

which vanishes in case of a good approximation.

A generic observable $\mathcal{O}$ is finally reweighted by computing during the data analysis

$$\langle \mathcal{O}\rangle = \langle \widehat{C}\rangle_\eta^{-1} \langle \mathcal{O}\widehat{C}\rangle_\eta. \tag{6.30}$$

Reweighting the observables allows to split the spectrum into one part being included in the update, while the correction factor C covers the rest.

6.4 Two-Pseudo-Fermion Fields

Like for the standard HMC we are interested to split the fermionic force by introducing (at least) a second pseudo-fermion field. Applying the Hasenbusch-trick as before by adding a shift ρ to the Dirac-Wilson operator seems not to be promising because this contradicts the idea of choosing best suited scaled and translated Chebyshev polynomials.

Instead we propose to follow the idea presented in [52] and introduce a smaller hopping parameter $\kappa_1 < \kappa$ to define[2] $M_1 = \mathbb{1} - \frac{\kappa_1}{\kappa}\hat{K}$ and split the determinant of $\hat{M}^\dagger\hat{M}$ according to

$$\begin{aligned}
\det\{\hat{M}^\dagger\hat{M}\} &= \det\{M_1^\dagger M_1 M_1^{-1} M_1^{\dagger -1}\hat{M}^\dagger\hat{M}\} \\
&= \det\{M_1^\dagger M_1\}\cdot\det\{M_1^{\dagger -1}\hat{M}^\dagger\hat{M}M_1^{-1}\} \\
&= \det\left\{\left[M_1 P_{1|n}\right]^\dagger\left[M_1 P_{1|n}\right]\right\}\left[\det\left\{P_{1|n}^\dagger P_{1|n}\right\}\right]^{-1} \\
&\quad\cdot\det\left\{\left[M_2 P_{2|n}\right]^\dagger\left[M_2 P_{2|n}\right]\right\}\left[\det\left\{P_{2|n}^\dagger P_{2|n}\right\}\right]^{-1},
\end{aligned} \tag{6.31}$$

with $M_2 = M_1^{-1}\hat{M}$. This gives rise to the two pseudo-fermion fields ϕ_1 and ϕ_2 both "living" on odd sites only

$$\begin{aligned}
\det\{\hat{M}^\dagger\hat{M}\} \propto \int &\mathcal{D}\phi_1^\dagger\,\mathcal{D}\phi_1\,\mathcal{D}\phi_2^\dagger\,\mathcal{D}\phi_2\;\exp\left\{-\phi_1^\dagger\left(M_1 M_1^\dagger\right)^{-1}\phi_1\right\} \\
&\cdot\exp\left\{-\phi_2^\dagger\left(M_1^\dagger\hat{M}^{\dagger -1}\hat{M}^{-1}M_1\right)\phi_2\right\}.
\end{aligned} \tag{6.32}$$

[2]Remember, $\hat{K}$ is proportional to κ. Hence alternatively the additional multiplication can be saved by defining $\hat{K}_1 \propto \kappa_1$.

With $0 \leq \varrho = \frac{\kappa_1}{\kappa} \leq 1$ and $\sigma = 1 - \varrho$, the first is governed by

$$M_1 = \mathbb{1} - \varrho \hat{K} = \varrho \hat{M} + \sigma \, \mathbb{1}, \tag{6.33}$$

while introducing M_2 for the second we find

$$M_2^{-1} = \hat{M}^{-1} M_1 = \varrho \, \mathbb{1} + \sigma \hat{M}^{-1}. \tag{6.34}$$

The advantage of this splitting becomes obvious if one considers e.g. the condition number[3]: M_1 has (for typical gauge configurations) a lower condition number than $\hat{M}$ since $\kappa_1 < \kappa$, while the condition number of M_2 is essentially the same as for $\hat{M}$. Due to a MTS integration scheme the expensive part can be computed less often and one thus profits. Moreover, this contribution gets suppressed by the factor $\sigma < 1$ Next we seek polynomial expressions for M_1^{-1} and M_2^{-1} in terms of scaled and translated Chebyshev polynomials.

For $P_{1|n}(M_1) \approx M_1^{-1}$ the original relations (6.7) and (6.8) are modified by replacing $\widetilde{K}$ with $\varrho\widetilde{K}$ and scaling $\hat{\delta}$, $\hat{e}$ accordingly i.e. $\hat{\delta} \to \hat{\delta}_1 = \varrho\hat{\delta}$ and $\hat{e} \to \hat{e}_1 = \varrho\hat{e}$. With $\hat{e}_1$ and $d_1 = 1 + \varrho\hat{\delta}$ we get new coefficients

$$a_{1|n} = \left(d_1 - \hat{e}_1^2 a_{1|n-1}/4\right)^{-1} \quad \text{with} \quad a_{1|1} = \left(d_1 - \hat{e}_1^2/(2d_1)\right)^{-1}, \tag{6.35}$$

and the recursions are

$$\begin{aligned} R_{1|n+1}(M_1) &= a_{1|n}\varrho\widetilde{K} R_{1|n}(M_1) + (1 - d_1 a_{1|n}) R_{1|n-1}(M_1) \\ &\qquad \text{with} \quad R_{1|1} = \varrho\widetilde{K}/d_1 \quad \text{and} \quad R_{1|0} = \mathbb{1}; \qquad (6.36) \\ P_{1|n}(M_1) &= a_{1|n}(\mathbb{1} + \varrho\widetilde{K} P_{1|n-1}) + (1 - d_1 a_{1|n}) P_{1|n-2}(M_1) \\ &\qquad \text{with} \quad P_{1|1} = a_{1|1}(\mathbb{1} + \varrho\widetilde{K}/d_1) \quad \text{and} \quad P_{1|0} = \mathbb{1}/d_1. \qquad (6.37) \end{aligned}$$

Furthermore, the variation $\delta\{\widetilde{K}\}$ in (6.20) gets multiplied by ϱ.

In case of the second pseudo-fermion only the inverse term in (6.34) is polynomially approximated re-using the expressions obtained for one pseudo-fermion ((6.7) - (6.9)). Hence M_2^{-1} becomes with $P_n(\hat{M}) \approx \hat{M}$

$$M_2^{-1} \approx P_{2|n} = \varrho + \sigma P_n(\hat{M}). \tag{6.38}$$

The constants $\hat{e}$, $\hat{\delta}$ and d remain unchanged as well.

With these approximations at hand we consider next the three computational parts: generation of the pseudo-fermions, calculating the bosonic forces

[3]For non-normal matrices A we refer by the term condition number to the square root of the condition number of $A^\dagger A$.

and estimating the correction factors. The first pseudo-fermion is essentially a heavier version of the original pseudo-fermion, easier to be approximated and the equations are only modified trivially. Hence we draw our attention right away to the second pseudo-fermion. ϕ_2 is generated by

$$\phi_2 = (\varrho + \sigma P_n)^{-1} \eta_2 = (M_1 - \sigma R_{n+1})^{-1} \hat{M} \eta_2. \tag{6.39}$$

The inversion appearing in (6.39) is very expensive since the $\mathrm{cond}(M_1) \gg \mathrm{cond}(\mathbb{1})$ (if comparing with (6.11)) and at each iteration step the entire polynomial R_{n+1} has to be applied. Therefore we have to cast it such that a good initial guess provided to the CG ensures that only a few iterations are performed. Introducing $\mathbb{1} = M_1 M_1^{-1}$ and writing (6.39) in an appropriate form of a normal equation we find

$$\begin{aligned} \phi_2 &= \left[(M_1 - \sigma R_{n+1})^\dagger (M_1 - \sigma R_{n+1}) \right]^{-1} (M_1 - \sigma R_{n+1})^\dagger M_1 \tilde{\eta} \\ \tilde{\eta} &= M_1^{-1} \hat{M} \eta_2. \end{aligned} \tag{6.40}$$

Assuming now a polynomial of sufficiently high degree the approximation is very good and hence R_{n+1} vanishes. Hence $\tilde{\eta}$ is a good initial guess for the inversion containing R_{n+1}. Obviously, this guess comes not for free: to create $\tilde{\eta}$ we still have to invert M_1 but we decoupled this inversion[4] from the polynomial of degree n.

Looking next at the calculation of the bosonic force for the second pseudo-fermion we find fortunately only little changes. The computation gets minimally changed by substituting $\xi_n^\dagger \to \sigma(\sigma \xi_n^\dagger + \varrho \phi_o^\dagger)$ in (6.19). Here we profit that the constant term ϱ in (6.38) does not contribute itself to the variation.

Finally, we focus at the correction factor. For two pseudo-fermions we can either estimate both expressions in (6.31) separately or in a combined approach which we present here. The combined approach introduces less noise and allows to retrieve comparable information without significant additional costs. Hence we estimate generalizing (6.24) to two pseudo-fermions with

$$\begin{aligned} \widehat{C} = \exp \Bigg\{ \eta^\dagger \Big[\mathbb{1} - M_1^\dagger (M_1 - \sigma R_{n+1})^{\dagger -1} \left(\mathbb{1} - R_{1|n+1} \right)^{\dagger -1} \\ \times \left(\mathbb{1} - R_{1|n+1} \right)^{-1} (M_1 - \sigma R_{n+1})^{-1} M_1 \Big] \eta \Bigg\}, \end{aligned} \tag{6.41}$$

[4] Depending on the choice of ϱ inverting M_1 may be significantly easier than the original problem of inverting $\hat{M}$.

where we rearranged the determinants in (6.31) accordingly. (6.41) is computed by the following steps

$$\hat{\eta} = \left[(M_1 - \sigma R_{n+1})^\dagger (M_1 - \sigma R_{n+1}) \right]^{-1} (M_1 - \sigma R_{n+1})^\dagger M_1 \eta \tag{6.42}$$

$$\zeta = \left[\left(\mathbb{1} - R_{1|n+1} \right)^\dagger \left(\mathbb{1} - R_{1|n+1} \right) \right]^{-1} \left(\mathbb{1} - R_{1|n+1} \right)^\dagger \hat{\eta}, \tag{6.43}$$

which give rise to the combined correction factor $\widehat{C}$ as well as to $\widehat{C}_1$ and $\widehat{C}_2$, the correction factors for the approximation of each pseudo-fermion

$$\widehat{C} = \exp\left\{ \mathrm{Re}\left\{ \left(\eta^\dagger + \zeta^\dagger \right) (\eta - \zeta) \right\} \right\} \tag{6.44}$$

$$\widehat{C}_1 = \exp\left\{ \mathrm{Re}\left\{ \left(\hat{\eta}^\dagger + \zeta^\dagger \right) (\hat{\eta} - \zeta) \right\} \right\} \tag{6.45}$$

$$\widehat{C}_2 = \exp\left\{ \mathrm{Re}\left\{ \left(\eta^\dagger + \hat{\eta}^\dagger \right) (\eta - \hat{\eta}) \right\} \right\}. \tag{6.46}$$

Like for the heatbath of the second pseudo-fermion we require η in (6.42) to be a good initial guess for the CG to avoid very expensive iterations, which forces the same constraint viz. R_{n+1} must (almost) vanish.

In case of two pseudo-fermions $\widehat{C}$ is the quantity used for reweighting the observables. $\widehat{C}_1$ and $\widehat{C}_2$ provide only useful information to tune the algorithm, in particular to find the minimal degree of each polynomial such that the algorithm is working fine.

All in all the second pseudo-fermion forces two inversions - one at the beginning, the other at the end of a trajectory - which can become prohibitively expensive if the degree of the polynomial is too small. Selecting an appropriate degree only one standard inversion at the beginning of the trajectory remains. The force computation remains cheap where normally most time is spent.

A generalization to multiple pseudo-fermions is straight forward (see Ref. [54]) but seems not to be very promising due to the complicated nature of the second and subsequently introduced pseudo-fermions.

6.5 Choosing Polynomial Parameters

Before testing the above described algorithm the following kind of parameters need to be determined:

- Parameters characterizing the bounding ellipse of $\hat{M}$'s spectrum, namely the shift along the real axis $d = 1 + \hat{\delta}$ and the eccentricity $\hat{e}$.
- The degree n of the polynomial specifying the quality of the approximation.

- If employing two pseudo-fermions by the Hasenbusch-Trick we have to specify in addition the ratio $\varrho = \kappa_1/\kappa$ making the first pseudo-fermion heavier than the standard one pseudo-fermion.

The first task is already addressed in Chapter 4. Given a set of configurations we can determine the optimal $\hat{\delta}$ and $\hat{e}$ by computing a set of Lanczos eigenvalues and then seek the optimal ellipse (cf. Section 4.2.2). If no configurations at the desired physical parameters are available we propose to start the thermalization with $\hat{\delta} = \hat{e} = 0.35$ which according to Table 4.7 seems to be a save and reasonable choice. Of course here it is advantageous that the spectrum of the symmetric even-odd preconditioned operator shows only little dependence on the physical parameters. Later one may optimize $\hat{\delta}$ and $\hat{e}$ by computing the Lanczos eigenvalues on the then generated configurations or do fine tuning with the NPHMC algorithm.

Finding the optimal degree of the polynomial is crucial to achieve a good performance of the algorithm. If the degree of the polynomial is too large the approximation becomes exact but the algorithm slow; while choosing the degree too small makes the approximation imprecise and the noise introduced by the correction factor becomes too influential. A strategy to find a good choice on n is to start the update by choosing n equal to half the average iteration number required by a CG to invert the given Dirac operator.[5] This value is either known or can be guessed from other simulations. A first check if the degree n is sufficient enough is to compute the mean value of the correction factor. By construction it approaches 1 for a perfectly approximating polynomial. Nevertheless a significant deviation does not matter as long as it remains, let us say, in $[0.5, 2]$. A more suitable measure of the quality of the approximation is the variance of the correction factor. Considering the normalized quantity ς_C we choose the degree n of the polynomial such that ς_C is sufficiently small. In Chapter 8 we try to find an upper bound on ς_C.

Effectively, the degree of the polynomial enters at two distinct parts of the computation as outlined in Section 5.3. The above discussed criteria are restrictions for the degree of the polynomial used in the *acceptance* Hamiltonian. For the *guidance* Hamiltonian one can use in principle a polynomial of different degree. However, choosing this degree too small, the integration of the equation of motion deviates from the acceptance Hamiltonian, hence ΔH becomes large and the acceptance low. Integrating finer by a degree of the polynomial greater than the one used in the acceptance Hamiltonian is probably only of little use.

[5] The factor 2 arises from the fact that a CG inverts $\hat{M}\hat{M}^\dagger$, while we have to specify the degree of the polynomial approximating $\hat{M}^{-1}$.

To specify the ratio ϱ is last remaining task which also depends on the choice of the integrator and its parameters. Motivated by [77] it seems to be favorable to choose the timescales according to the force contribution. Hence there may not be one optimal value for ϱ. Once this ratio is fixed, the polynomial parameters $(d,\, \hat{\delta},\, \hat{e})$ can be scaled as mentioned in Section 6.4, while adjusting the degree of the polynomials will require to repeat the previously mentioned procedure. Moreover, there arises a lower bound for the degree of the approximating polynomial of the second pseudo-fermion. Here one is forced to choose n at least so that very expensive inversions are avoided.

A detailed discussion on how to find a good choice on the degree of the polynomials and the ratio ϱ is postponed to Chapter 8. Analyzing the performance of the algorithm we give a practical method taking into account the computational costs of the pseudo-fermions. These costs are estimated by counting the number of applications of the Dirac-Wilson operator to perform one trajectory including the generation of the pseudo-fermion in the heatbath and the computation of the correction factor (but neglecting anything else). Moreover, we assume an equal degree polynomial (n_i) in the heatbath and the force computation, perform s_i integration steps (= number of times the force is computed) and give the number of CG iterations by e.g. $\#(\mathrm{CG}_{\mathrm{HB}})_i$, where HB stands for the pseudo-fermion heatbath and CF the computation of the correction factor. For the first pseudo-fermion we find

$$\mathcal{C}_1 = 2 \cdot n_1 \left(1 + s_1 + \#(\mathrm{CG}_{\mathrm{HB}})_1 + \#(\mathrm{CG}_{\mathrm{CF}})_1\right) \tag{6.47}$$

and including the number of CG iterations to invert M_1 we have for the second pseudo-fermion

$$\mathcal{C}_2 = 2 \left[n_2 \left(1 + s_2 + \#(\mathrm{CG}_{\mathrm{HB}})_2 + \#(\mathrm{CG}_{\mathrm{CF}})_2\right) + 1 + \#(\mathrm{CG}_{\mathrm{M}})\right]. \tag{6.48}$$

Both cost figures refer to the simpler computation of the force with higher memory consumption. Reducing the memory requirements, the force computation becomes a factor 2 more expensive thus s_i is replaced by $2s_i$.

Chapter 7

Improvements and Performance of the Standard HMC

In this chapter we present improvements to the standard HMC algorithm used by the Alpha-Collaboration[1]. Furthermore, we show some algorithmic properties found in the course of two-flavor dynamical fermion simulations. These simulations are part in Alpha's long term project to determine the QCD Λ-parameter starting from experimental low energy hadronic input and using perturbation theory only in a renormalized coupling at sufficiently high energy scales. At these energy scales it was demonstrated that perturbation theory is very accurate. For two-flavor QCD the Λ-parameter is computed in units of the low energy scale $L_{\max}$ [6] using the Schrödinger functional scheme. This scale is defined implicitly by the renormalized coupling taking in that scheme the particular value $\bar{g}^2_{\rm SF}(\mu = 1/L_{\rm ren}) = 4.61$. Determining now $\Lambda_{SF} L_{\max}$ we get a continuum, universal result and the relation between $\Lambda_{\rm SF}$ and $\Lambda_{\overline{\rm MS}}$ is known exactly.[79] The remaining task is to replace $L_{\max}$ by an experimentally accessible low-energy scale like the kaon decay constant F_K. (Currently, the last step is substituted by using the chirally extrapolated Sommer reference scale r_0.[6, 80])

In the standard HMC we use even-odd preconditioning and employ the Hasenbusch-trick as discussed in Section 5.5.1. Switching from asymmetric to symmetric even-odd preconditioning (see Chapter 3) proves to be advantageous for the Hermitian operator, too. In the following section we discuss our findings and show how one profits by using different time scales for the integration with two pseudo-fermions (MTS).

The second section is devoted to performance studies of our HMC algorithm incorporating the aforementioned improvements. In particular we

[1] `http://www-zeuthen.desy.de/alpha/`

focus on the dependence of autocorrelation times of observables on the trajectory length τ and look at the spectral gap as quantity reflecting the stability of the algorithm.

Finally, we report on the physical aspects of our large volume two-flavor QCD simulations and show a scaling test for different observables in the last section. The findings presented in this chapter are published in [7, 81–83].

7.1 Improvements

7.1.1 Symmetric vs. Asymmetric Even-Odd Preconditioning

Starting simple we compare first the effect of asymmetric versus symmetric preconditioning for one pseudo-fermion only. According to Chapter 3 we get different expressions for the contributions to the action namely for the factorized part of the determinant S_{det}, and the action of the pseudo-fermion, S_b. In order to obtain a measure on these quantities we follow an ansatz by Lüscher to compute the corresponding forces.[77] Here we refer by the term force to the traceless and anti-Hermitian part of the matrix (T.A.)

$$\mathcal{F}_\mu(x) \equiv 2\,[U_\mu(x)F_\mu(x)]_{\text{T.A.}} = \; i\,\dot{\Pi}_\mu(x) \tag{7.1}$$

and define the magnitude of the force to be a real number

$$\|\mathcal{F}\|^2 \equiv -\,2\,\text{Tr}\,\{\mathcal{F}^2\} = 2\,\text{Tr}\,\{\mathcal{F}F^\dagger\} \geq 0, \tag{7.2}$$

with $F_\mu(x)$ given by (5.10).[2] Computing the forces resulting from S_{det} and S_b, differences between both versions of preconditioning can be seen and are shown in Fig. 7.1. Factorizing for the symmetric version additionally M_{oo} almost doubles the widths of the distribution of the determinant force. However, this force remains much smaller than the bosonic force. Here we find a significant improvement for the symmetric version since this force distribution is shifted towards smaller values and has a width of roughly 2/3 compared to the one corresponding to the asymmetric version. Hence large forces causing larger energy violations (spikes) occur less often. Moreover, smaller forces allow to increase the step size of the HMC algorithm which leads to gain of about 10-30%. This is in agreement with findings by other groups [33, 54] as well as with our results presented for the non-Hermitian operator in Chapter 4.

[2]Unfortunately, there is no consistent use of the term "force" in the literature. A.D. Kennedy suggests to look instead at the corresponding Poisson brackets which provide a uniquely defined alternative.

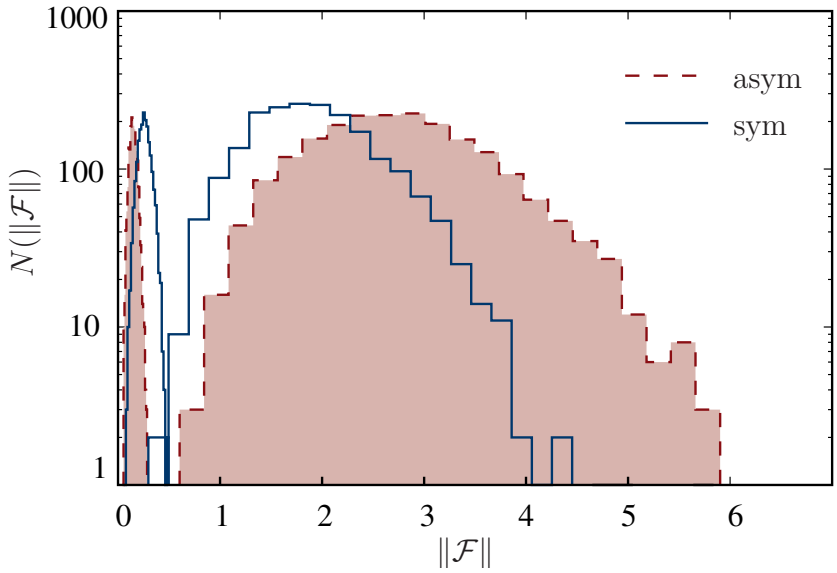

Figure 7.1. $\mathcal{F}_{\text{det}}$ narrow distribution at the left; $\mathcal{F}_b$ broad distributions. Computed on a 8^4 lattice at $\beta = 6.32$, $\kappa = 0.13577$ in the time slice $T/2$.

7.1.2 MTS Integration

Measuring the forces becomes furthermore useful when introducing by the Hasenbusch-trick several pseudo-fermions (see Section 5.5.1) and one seeks the "optimal" choice of parameters. Encouraged by the speedup of HMC simulations due to multiple time scale integration reported in [84] we switch from the standard Sexton-Weingarten integration scheme (cf. Section 5.4) to a MTS scheme. The idea of MTS integration is simple: based on the leap-frog integrator one introduces different times scales to compute expensive but small contributions less often than cheap but larger ones.[85][3]

Following Urbach et al. one seeks parameters ρ_i giving rise to several pseudo-fermions such that the more expensive contributions have a small share in the total force and can hence be computed less often. Since the costs occurring for a pseudo-fermion are dominated by the number of inversions required by the CG called in the course of the MD evolution one has to consider a cost figure combining both.[4] Apparently, this is a non-trivial tuning problem. Therefore we restrict ourselves to two pseudo-fermions with one parameter ρ and plot the magnitude of the force and the number of CG iterations in dependence of ρ in Fig. 7.2 (lower plot).

[3]Originally, this idea is motivated by separating the "UV- and the IR part" of the pseudo-fermion force by introducing a polynomial (see also [86]).

[4]For simplicity we assume that the autocorrelation is not affected by the choice ρ_i.

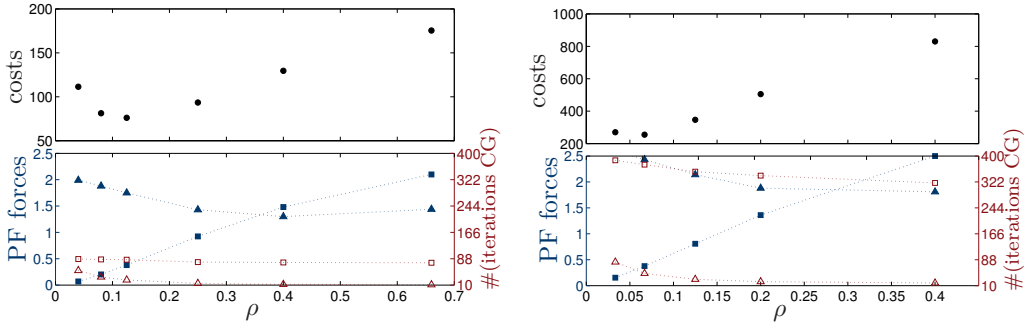

Figure 7.2. Bosonic forces (full symbols) and CG iteration numbers (open symbols) for two pseudo-fermions computed on one 8^4 lattice gauge field configuration at $\beta = 6.32$, $\kappa = 0.13577$ (left) and one 16^4 configuration at $\beta = 5.2$, $\kappa = 0.13568$ (right) both in the time slice $T/2$. The upper panels show our estimated cost figure.

One finds that the forces show a much stronger dependence on ρ than the numerical costs given by the average CG iteration number. Moreover, the costs of the second pseudo-fermion (boxes) are dominatingly larger than the ones for the first pseudo-fermion (triangles).

Neglecting any additional overhead and keeping all other parameters fixed we consider the costs of the pseudo-fermion action to be proportional to

$$\mathcal{C}(\mathrm{PF1}, \mathrm{PF2}) \propto \sum_{i=1}^{2} \langle \|\mathcal{F}_i\| \rangle \cdot \#(\text{iterations CG})_i \tag{7.3}$$

and seek ρ yielding the minimum (see Fig. 7.2, upper plots). The data presented are computed for one thermalized configuration: left 8^4 lattice at $\beta = 6.32$, $\kappa = 0.13577$ and right 16^4 lattice at $\beta = 5.2$, $\kappa = 0.13568$. For the 8^4 lattice we find the optimal ρ to be 0.125, while on the larger lattice $\rho_{\mathrm{opt}} = 0.06688$. Moreover, the costs increase much faster thus a good choice on ρ becomes very important. A rule of thumb on how to choose ρ is unfortunately not obvious. However it seems to be good choice to use the magnitude of the smallest eigenvalue $\lambda_{\min}(\hat{Q})$ as lower bound and start seeking an optimal ρ around $5 \cdot \lambda_{\min}$. With an optimal ρ MTS integration leads to a gain of up to 50% compared to the Sexton-Weingarten integrator. Introducing more than two pseudo-fermions does not seem to promising because this does not decrease the costs for the expensive fermion while its force is already small.

Possibly a little further gain can be achieved by employing the Hasenbusch-trick analogously as presented in Section 6.4 for the NPHMC since this allows the second pseudo-fermion to stay "symmetric" [54].

7.2 Performance

7.2.1 HMC dependence on trajectory length

The length τ of a HMC trajectory is a parameter for all HMC-type algorithm but receives commonly little attention and mostly $\tau = 0.5$ or 1.0 is chosen. Recalling the features of the HMC algorithm we pick at the beginning of the trajectory the Gaussian momenta and pseudo-fermions while at the end we perform an accept-/reject-step to make the algorithm exact. Hence if one traverses the same distance in phase-space once with a shorter, once with a longer trajectory (keeping the step-size $\delta\tau$ fixed), less noise will be introduced in the latter case but larger energy and reversibility violations can occur. To explore the influence of the trajectory length τ we monitor algorithmic quantities and compute in addition autocorrelation times for some observables. We perform three kind of simulations

- in the pure gauge theory
- in quenched QCD
- in two-flavor QCD

keeping always for one set of parameters the rate of acceptance (almost) constant. In particular we look at the average squared energy violations $\langle \Delta H^2 \rangle$, compute the average reversibility violations, $\langle |\Delta H|_{\leftrightarrow} \rangle$, by flipping the sign of the momenta after performing a trajectory and integrate backwards to compute the difference. The simulation parameters and results are summarized in Appendix D.1.

In case of the pure gauge theory (cf. Tab. D.1) we monitor the autocorrelation of $\partial S/\partial \eta$ as introduced in [87]. In contrast to the plaquette, $\partial S/\partial \eta$ is an observable dominated by long-distance fluctuations and has typically an autocorrelation time larger than one. The inverse of its expectation values defines the Schrödinger functional renormalized coupling. Comparing the effect of four different choices for the trajectory length, $\tau = 1/2, 1, 2, 4$, one sees in a direct comparison of the autocorrelation function $\rho(t)$ advantages for $\tau = 2, 4$: at shorter time separation ρ is much smaller and shows moreover a non-monotonic behavior like it was observed previously for hybrid overrelaxation algorithms. Integrating the autocorrelation function results for a simulation run of given MD-time length and measurement frequency in the *integrated autocorrelation time* τ_{int} which is inversely proportional to the error squared of the observable (cf. Appendix C.2 for a detailed description). In practice the integration is replaced by a sum and a window must be chosen, where to stop the summation. Following the prescription given in [88]

this is automatized balancing statistical against systematic errors. Besides these values we list in Tab. D.1 truncated integrated autocorrelation times, where we fixed the summation window to 25. Typically 80% of the true τ_{int} are by then accumulated and the uncertainty is almost half of that of τ_{int}. Comparing the truncated values with identical summation window, we find this quantity to be minimal for $\tau = 2$ independent of the considered lattice spacings as can be seen in Fig. 7.3. The substantial variation of roughly a factor two translates directly into a corresponding speedup of simulations whose costs are dominated by the HMC. Checking otherwise the dependence of the

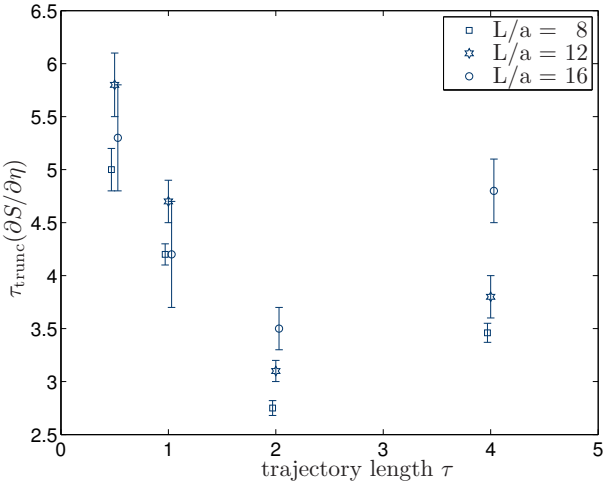

Figure 7.3. Truncated integrated autocorrelation times of $\partial S/\partial \eta$ for pure gauge theory in dependence of the trajectory length τ at different lattice spacings, $L = 0.7fm$.

algorithmic observables on τ we see only a moderate dependence of $\langle \Delta H^2 \rangle$. Being in a regime of high acceptance P_{acc} we confirm $\langle \Delta H^2 \rangle \simeq 2\pi(1 - P_{\mathrm{acc}})$ [8]. Concerning the reversibility violations we find $\langle |\Delta H|_{\leftrightarrow} \rangle$ grows as $\sqrt{\tau}$ or even slower. Hence we conclude that $\tau = 2$ is the superior choice for the pure gauge systems studied.

Turning next to quenched QCD we study one $8^3 \times 32$ lattice at $\beta = 6.0$ and $\kappa = 0.1388$ with $\tau = 1/2, 2, 4$ and run length $t_{\mathrm{run}} = 4 \cdot 8000$. As a first step to two-flavor QCD simulations we focus at the fermionic correlators, $f_A(x_0)$ and $f_P(x_0)$, which corresponds to the propagation of a quark and an anti-quark from a boundary to a point in the bulk of the lattice, where the axial current or the pseudo-scalar density annihilates them [89] and at f_1, the amplitude of the boundary-to-boundary propagation through the lattice. f_1 serves to normalize the other correlators and far from the boundaries we have $Z_A f_A(x_0)/\sqrt{f_1} \sim F_{PS}\, \mathrm{e}^{-(x_0 - T/2)M_{PS}}$ with F_{PS} the decay constant and M_{PS} the pseudo-scalar mass.

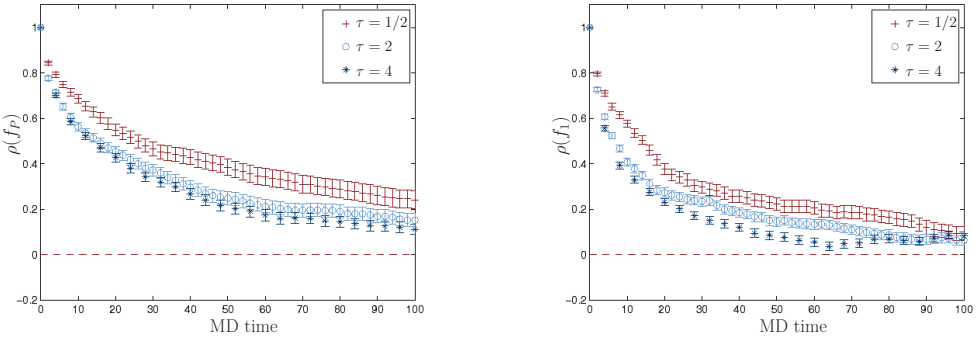

Figure 7.4. Normalized autocorrelation function for the correlators f_P (left) and f_1 (right) in quenched QCD.

The autocorrelation functions for f_P and f_1 are plotted for the three τ considered in Fig. 7.4 and the corresponding data can be found in Tab. D.2. Again $\tau = 2, 4$ are superior to the length $1/2$ leading to a τ_{int} reduced by a factor of about 2. Varying τ there is almost no change in the acceptance, while the reversibility violations $\langle |\Delta H|_{\leftrightarrow} \rangle$ scales like $\sqrt{\tau}$.

Finally, we study the same observables in two-flavor QCD where we employ the algorithm as discussed in the previous section. The spatial volume is of $(2\,\text{fm}^3)$; the quark mass is around the strange quark m_s. On a $24^3 \times 32$ lattice we compute for two different τ values each a run of length $2 \cdot 2000$. The data for this simulations can be found in Tab. D.3 and the corresponding plots for the autocorrelation function are shown in Fig. 7.5. Now the errors have grown significantly as expected. For both τ values we have practically the same rate of acceptance and see that for $\tau = 2$ the drop off of the auto-

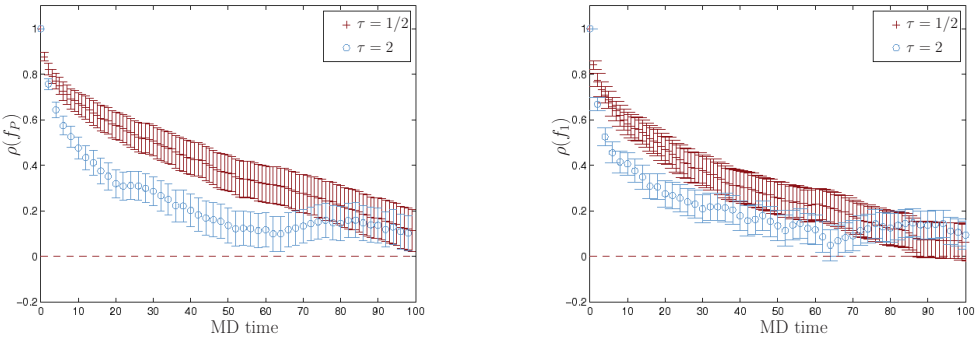

Figure 7.5. Normalized autocorrelation function for the correlators f_P (left) and f_1 (right) in two-flavor QCD.

correlation function is improved. Looking at truncated autocorrelation times confirms this as well as looking at the derived quantities F_{PS} and M_{PS}.

After looking at autocorrelation times for particular observables we emphasize different observables of the same simulation can have quite different autocorrelation times. We compile observed integrated autocorrelation times τ_{int} for five quantities discussed and defined subsequently in Tab. D.6.

7.2.2 Spectral Gap

As mentioned in the introduction, large volume simulations are the challenges of today. As can be seen in Fig. 1.1 large energy violations can occur rather frequently and lower the performance of HMC simulations. In Ref. [90] Del Debbio et al. point out that these energy violations are related to tiny eigenvalues of the Dirac-Wilson operator and define μ, the spectral gap of the Hermitian Dirac-Wilson operator, as tool to diagnose the stability of the HMC algorithm. Transferring this to our symmetrically even-odd preconditioned Dirac-Wilson operator in the Schrödinger Functional we define

$$\hat{\mu} = \frac{1}{4\kappa\hat{c}_0}\sqrt{\lambda_{\min}}, \tag{7.4}$$

where $\lambda_{\min}$ is the minimal eigenvalue of $\hat{Q}^S\hat{Q}^{S\dagger}$ as defined in (3.15) additionally multiplied by $\hat{c}_0 = (1+64\kappa^2)^{-1}$. The normalization in (7.4) is chosen such that it is given by the quark mass in the free theory with periodic boundary conditions. Since only $\hat{Q}^S$ can lead potentially to unbounded fluctuations of molecular dynamics forces, a sufficient condition for the stability of the HMC algorithm arises if the distribution of $\hat{\mu}$ is well separated from the origin. Computing the lowest eigenvalues of $\hat{Q}^S\hat{Q}^{S\dagger}$ by the Kalkreuther-Simma algorithm [91] we obtain $\hat{\mu}$ and plot the binned distribution for simulation runs C_1 and C_2 (cf. Appendix D.2) in Fig. 7.6.

In both volumes we find a clear gap of the median from the origin. But in a few cases during the simulation eigenvalues as small as a third of this value occur. Furthermore, one sees that increasing the volume shifts the distribution to smaller values leading also to a significant difference in the medians. Considering next the variance $\hat{\sigma}^2$ as measure on the width of the μ distribution we confirm the claim in [90] that $a\sigma\sqrt{L^3T/a^4} \approx$ const. For simulations A_1, C_1 and C_2 we find

$$\hat{\sigma}\sqrt{L^3T}/a = \begin{cases} 1.437(64) & A_1 \\ 1.268(23) & C_1 \\ 1.477(33) & C_2 \end{cases} \tag{7.5}$$

which varies only by about 15%.

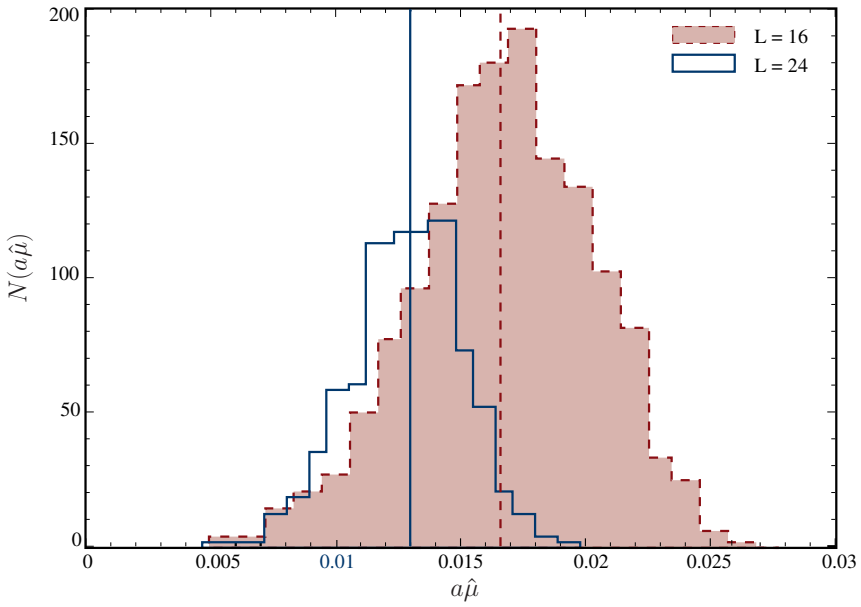

Figure 7.6. Histogram of $\hat{\mu}$ for two different spatial volumes (simulations C_1 and C_2) with the lines indicating the medians.

7.3 Scaling Test

In this section we investigate cutoff effects on a number of non-perturbatively renormalized quantities. In order to keep systematic effects due to a varying volume negligible, we compare a series of simulations in a fixed (but quite large) volume on a physical scale (for parameters see Tab. D.4 and D.5). More precisely we determine $L/L^* = 3.00(4)$, $3.07(3)$ and $T/L^* = 3.93(4)$, $4.09(3)$ on the A and B lattices. At $\beta = 5.2$, the volumes came out less uniform, $L(C_1)/L^* = 2.46(5)$, $L(C_2)/L^* = 3.69(6)$ and $T(C_i)/L^* = 4.92(10)$. We discuss how to correct for these small mismatches after introducing the finite volume observables of this study.

They are extracted from the zero spatial momentum boundary-to-bulk correlation functions, $\mathrm{f_A}(x_0)$, $\mathrm{f_P}(x_0)$ in the pseudo-scalar channel, $k_\mathrm{V}(x_0)$ in the vector channel and the boundary-to-boundary pseudo-scalar correlator f_1 [89]. We include the $O(a)$ improvement term proportional to $\mathrm{c_A}$ [92] in $\mathrm{f_{A,I}} = \mathrm{f_A} + a\,\mathrm{c_A}\,\partial_0\,\mathrm{f_P}$. Effective masses and decay constants

$$\mathrm{m}^{\mathrm{A}}_{\mathrm{eff}}(x_0) \equiv -\tfrac{1}{2}(\partial_0^* + \partial_0)\log(\mathrm{f_{A,I}}(x_0)) \tag{7.6}$$

$$\mathrm{m}^{\mathrm{P}}_{\mathrm{eff}}(x_0) \equiv -\tfrac{1}{2}(\partial_0^* + \partial_0)\log(\mathrm{f_P}(x_0)) \tag{7.7}$$

$$\mathrm{m}^{\mathrm{V}}_{\mathrm{eff}}(x_0) \equiv -\tfrac{1}{2}(\partial_0^* + \partial_0)\log(k_\mathrm{V} V(x_0)) \tag{7.8}$$

$$\begin{aligned}
\mathrm{F_{eff}}(x_0) &\equiv -2Z_A \frac{\mathrm{f_A}(x_0)\,(1+b_\mathrm{A} am_\mathrm{q})\exp(\mathrm{m^A_{eff}}(x_0)(x_0-T/2))}{\left(f_1\ \mathrm{m^A_{eff}}(x_0)\,L^3\right)^{1/2}} \\
&= -2Z_A\,(1+b_\mathrm{A} am_\mathrm{q})\frac{\mathrm{f_{A,I}}(T/2)}{\left(f_1\ \mathrm{m^A_{eff}}(T/2)\,L^3\right)^{1/2}} \quad \text{at} \quad x_0 = T/2 \qquad (7.9) \\
\mathrm{G_{eff}}(x_0) &\equiv 2Z_p\,(1+b_\mathrm{P} am_\mathrm{q})\frac{\mathrm{f_P}(x_0)\exp(\mathrm{m^P_{eff}}(x_0)(x_0-T/2))\ \mathrm{m^P_{eff}}(x_0)^{1/2}}{\left(f_1\,L^3\right)^{1/2}} \\
&= 2Z_P\,(1+b_\mathrm{P} am_\mathrm{q})\frac{\mathrm{f_P}(T/2)\ \mathrm{m^P_{eff}}(T/2)^{1/2}}{\left(f_1\,L^3\right)^{1/2}} \quad \text{at} \quad x_0 = T/2 \qquad (7.10)
\end{aligned}$$

are related to (L-dependent) masses and matrix elements,

$$\begin{aligned}
&\mathrm{m^A_{eff}}(x_0) \approx \mathrm{M_{PS}} \approx \mathrm{m^P_{eff}}(x_0)\,, \quad \mathrm{m^V_{eff}}(x_0) \approx \mathrm{M_V}\,, \\
&\mathrm{F_{eff}}(x_0) \approx \mathrm{F_{PS}}\,, \quad \mathrm{G_{eff}}(x_0) \approx \mathrm{G_{PS}}\ . \qquad (7.11)
\end{aligned}$$

These relations hold in the limit of large x_0 and T up to correction terms [89]

$$\begin{aligned}
O_\mathrm{eff}(x_0) = O + \eta_O \exp(-(E_1 - \mathrm{M_{PS}})\,x_0) + \tilde{\eta}_O \\
\exp(-E_2\,(T-x_0)) + \ldots\,, \qquad (7.12)
\end{aligned}$$

where the coefficients η_O and $\tilde{\eta}_O$ are ratios of matrix elements, E_1 is the energy of the first excitation in the zero momentum pion channel and E_2 in the vacuum channel. For not too small L and not too large $\mathrm{M_{PS}}$ we expect $E_1 \approx 3\,\mathrm{M_{PS}}$ and $E_2 \approx 2\,\mathrm{M_{PS}}$. Our results for the effective observables at $x_0 = T/2$ are listed in Tab. D.7 together with the bare current quark mass m stabilized by averaging over $T/3 \le x_0 \le 2T/3$,

$$\begin{aligned}
m &= \frac{1}{n_2 - n_1 + 1} \sum_{x_0/a=n_1}^{n_2} m(x_0)\,, \quad n_1 \ge T/3a\,,\ n_2 \le 2T/3a \qquad (7.13) \\
m(x_0) &= \frac{\frac{1}{2}(\partial_0^* + \partial_0) f_\mathrm{A}(x_0) + \mathrm{c_A}\, a\partial_0^*\partial_0 f_\mathrm{P}(x_0)}{2 f_\mathrm{P}(x_0)}\,. \qquad (7.14)
\end{aligned}$$

The results at $\beta = 5.3$ can be compared directly to those of [93], shown in Tab. D.8, for which the correction terms in eq. (7.12) can safely be neglected. In other words they correspond to $x_0, T \to \infty$. This allows us to estimate the effects due to $T(C) > T(A) \approx T(B)$ in addition to those coming from the mismatch in L.

1. For the matrix elements $\mathrm{F_{eff}}\,, \mathrm{G_{eff}}$ no systematic differences between B and D lattices are visible. No correction due to T is necessary. We

just interpolate the C_1 and C_2 results in L to $L/L^* = 3$ using the Ansatz $a_1 + a_2\, L^{-3/2} e^{-\mathrm{M_{PS}}\, L}$, with $\mathrm{M_{PS}}$ the pion mass on the larger volume. A small systematic error is added linearly to the statistical one. It is estimated by comparing with the result from an alternative interpolation with $a_1' + a_2'\, L^{-1}$.

2. We observe $|\,\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}(B)/\,\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}(D) - 1| \leq 0.03$ without a systematic trend as a function of the quark mass. We take this into account as a systematic error of 2% on $\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}(C)$ and[5] subsequently we interpolate in L as in 1. The numbers for $\mathrm{m}_{\mathrm{eff}}^{\mathrm{A}}$ are not used further.

3. Finite T effects are not negligible in the vector mass ($\mathrm{m}_{\mathrm{eff}}^{\mathrm{V}}(B)/\,\mathrm{m}_{\mathrm{eff}}^{\mathrm{V}}(D) - 1 \approx -0.10 \ \ldots \ -0.03$). We thus first perform a correction for the finite T effects using fits to eq. (7.12) with $E_1 = 2(\mathrm{M_{PS}}^2 + (2\pi/L)^2)^{1/2}$, $E_2 = 2\,\mathrm{M_{PS}}$. A systematic error of 50% of this correction is included for the result. Next the finite L correction is performed as above.

The interpolated values are included in Tab. D.7 as "simulation" C_{I}. After these small corrections we are ready to look at the lattice spacing dependence of our observables. To this end the necessary renormalization factors are attached (with perturbative values for $b_{\mathrm{A}}, b_{\mathrm{P}}$ [94]) and we form dimensionless combinations by multiplying with L^*. At lowest order in the quark mass expansion (in large volume), one has $M_{\mathrm{PS}}^2 \propto m$. It is thus natural to consider $[\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}\, L^*]^2/[\bar{m}(\mu_{\mathrm{ren}})L^*]$ instead of the quark mass itself. We choose $\bar{m}$ renormalized non-perturbatively in the SF scheme at scale $\mu_{\mathrm{ren}} = 1/L_{\mathrm{ren}}$ where $\bar{g}^2(1/L_{\mathrm{ren}}) = 4.61$ [95]. The quantities considered are shown in Fig. 7.7 as a function of the dimensionless $[\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}\, L^*]^2$. At $\beta = 5.3$ we have a few quark-mass points. As a reference, these are locally interpolated in $[\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}\, L^*]^2$ with a second order polynomial. For masses lighter than in simulation B_2, the interpolation involves the lightest three masses and for heavier ones, it involves the heaviest three masses. The two-sigma bands ($\pm 2\sigma$) of these interpolations are depicted as dotted vertical lines. Our results at the other β-values are seen to be in agreement with these error bands, which are generally around 5%, but 10% for $[\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}\, L^*]^2/[\bar{m}(\mu_{\mathrm{ren}})L^*]$ after all errors are included. Even if the precision is not very impressive, large cutoff effects are clearly absent.

So far we have discussed the scaling of the ground state properties for a given symmetry channel. We now turn to the size of cutoff effects affecting *excited state* contributions to the correlators. Figure 7.8 compares the effective pseudo-scalar masses $\mathrm{m}_{\mathrm{eff}}^{\mathrm{A}}$ and $\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}$ in simulation A_1 and B_2. The

[5] From eq. (7.12) this finite T effect scales with $\exp(-\,\mathrm{M_{PS}}\, T)$, yielding a reduction of 3% by a factor $[1 - \exp(-\,\mathrm{M_{PS}}\, L^*)]$ when one considers the difference between $T \approx 5L^*$ and the target $T = 4L^*$.

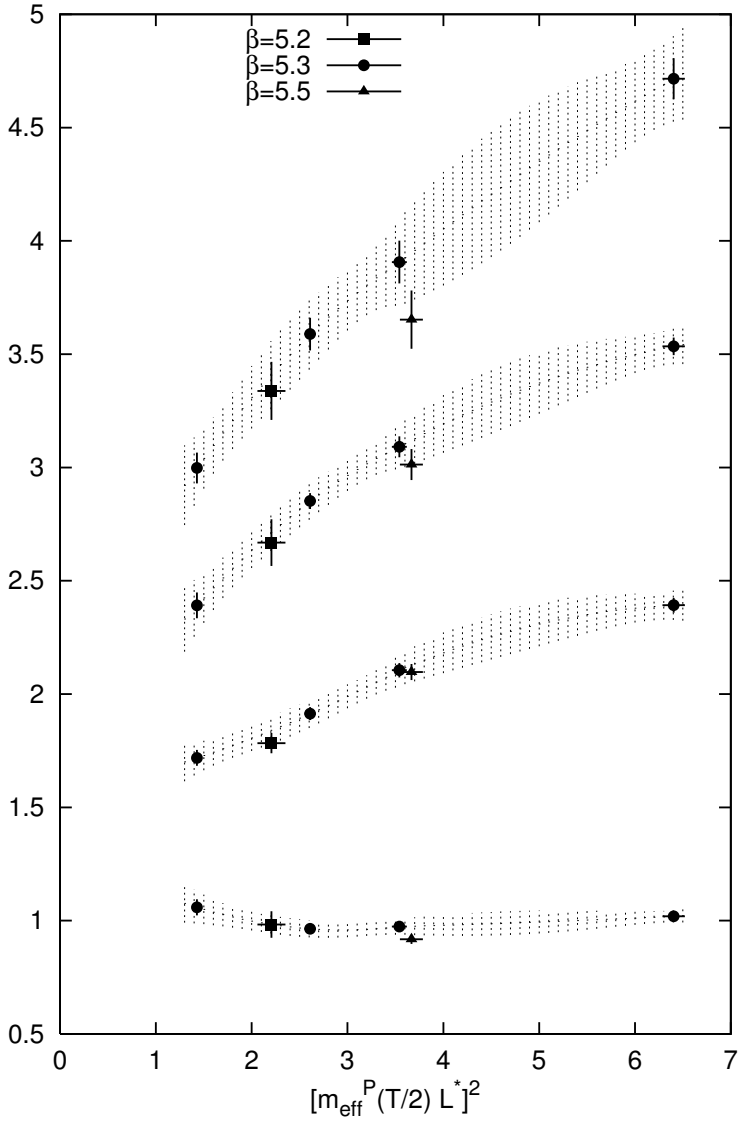

Figure 7.7. Dimensionless renormalized finite volume observables as a function of $[\mathrm{m}^{\mathrm{P}}_{\mathrm{eff}}\, L^*]^2$. From top to bottom $\mathrm{G}_{\mathrm{eff}}(L^*)^2$, $\mathrm{m}^{\mathrm{V}}_{\mathrm{eff}}\, L^*$, $4\,\mathrm{F}_{\mathrm{eff}}\, L^*$, $[\mathrm{m}^{\mathrm{P}}_{\mathrm{eff}}\, L^*]^2/[\bar{m}(\mu_{\mathrm{ren}})L^*]/15$ are shown. Squares, circles and triangle are for $\beta = 5.2\,,\ 5.3\,,\ 5.5$ respectively. Effective quantities are at $x_0 = T/2$. The dotted band is an interpolation of the $\beta = 5.3$ data as described in the text.

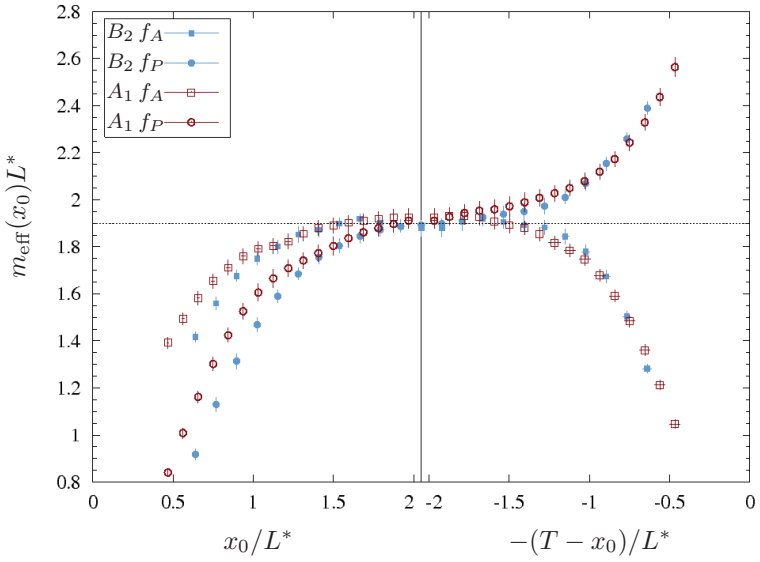

Figure 7.8. The effective pseudoscalar masses $\mathrm{m}_{\rm eff}^{\rm A}$ and $\mathrm{m}_{\rm eff}^{\rm P}$ in simulations B_2 and A_1. The horizontal error bars are shown on some of the points only for clarity. The horizontal line is to guide the eye. The vertical line indicates the middle of the B_2 lattice.

large size of the excited state contributions [83], while a drawback in extracting ground state properties, means that these functions are rather sensitive to the aforementioned cutoff effects. Because the A_1 time extent is shorter by 4(1)%, on this figure we have separately aligned the two boundaries of lattice A_1 and B_2. We observe that the two data sets are consistent within uncertainties well before the function flattens off. With the exception of $\mathrm{m}_{\rm eff}^{\rm P}$ for $x_0 < T/2$, the agreement sets in at a distance to the closest boundary of about L^*, where it is easily seen that several excited states contribute significantly to the correlation functions. Altogether this figure is evidence that the masses and matrix elements of the first excited state in both the pion and vacuum channel have scaling violations not exceeding the few percent level. But higher states can have rather significant discretization errors.

Chapter 8

Performance of the NPHMC

We start studying the performance of the NPHMC by investigating the dependence of the one pseudo-fermion algorithm on the polynomial degree n, the eccentricity $\hat{e}$ and the shift $\hat{\delta}$. Next we turn to the two pseudo-fermion version and seek optimal choices for the polynomial parameters and the ones specifying the integrator. Finally, we try to compare the performance of the NPHMC with our standard HMC, both with one and two pseudo-fermions.

8.1 Dependence on Polynomial Parameters

In order to obtain first performance figures and to test properties of the new non-Hermitian polynomial HMC algorithm we begin by exploring the dependence on the polynomial parameters for one pseudo-fermion. Starting from four independently thermalized configurations we generate $4 \cdot 100$ trajectories of length $\tau = 2$ for each data point. As setup we choose the same parameters as in run $S8_b$ (cf. Section 4.3.2), i.e. we have an 8^4 lattice at $\beta = 6.0$ and $\kappa = 0.13458$. Computing on a set of configurations generated by the standard HMC the optimal value of the eccentricity $\hat{e}$ and the real shift $\hat{\delta}$ we get: $\hat{e}_{\text{opt}} = 0.367$ and $\hat{\delta}_{\text{opt}} = 0.357$ for the $O(a)$-improved operator using symmetric even-odd preconditioning.

In the first test we use $\hat{e}_{\text{opt}}$ and $\hat{\delta}_{\text{opt}}$ varying only the degree of the polynomial n. We use the same degree polynomial for the generation of the pseudo-fermion, in the force computation as well as when computing the correction factor. As can be seen in Fig. 8.1 upper panel, the correction factor approaches 1 as the degree of the polynomial is increased. At the same time the number of CG iterations and our measure of the width of the correction factor's distribution (ς_C) drops (lower panel). For a reliable polynomial approximation we require the correction factor to fluctuate only moderately.

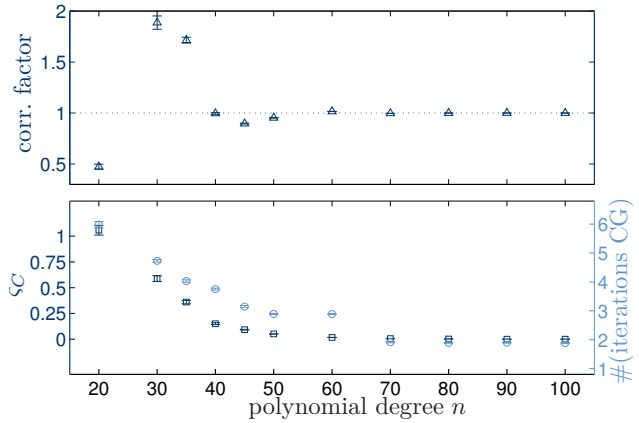

Figure 8.1. Correction factor C, its distribution ς_C and the number of CG iterations for the correction factor (CF) in dependence of the polynomial degree n.

Demanding $\varsigma_C < 0.15$ we guarantee the fluctuations are sufficiently small and have a lower bound on the degree of the polynomial. This bound is already reached for $n = 40$, a degree being $\sim 20\%$ smaller than half the average iteration number required by our standard HMC using the Hermitian operator. Choosing $n = 40$ we slightly deviate from importance sampling i.e. extremal eigenvalues of the Dirac-Wilson operator are not approximated by the polynomial. Hence they can not spoil the update, but they are taken into account by the reweighting factor to yield correct observables (cf. [58]).

From $n = 35$ to $n = 100$ the acceptance is almost the same (88% - 90%) but drops for $n = 20$ to 46%. Checking for reversibility violations as explained in the previous chapter, we find that these are independent of the polynomial degree n with an absolute value of $\langle|\Delta H|_{\leftrightarrow}\rangle$ of order 10^{-12} and a relative violation $\langle|\Delta H/H|_{\leftrightarrow}\rangle$ of order 10^{-16}. These findings are in perfect agreement with Aoki et al.[33]

Evaluating the cost figure $\mathcal{C}_1$, eq. (6.47), we observe that the numerical costs to perform one trajectory are strongly dominated by the number of integration steps m_1. For the discussed setup, a trajectory of length 2 is split into $m_1 = 50$ integrator steps of size $\delta\tau = 0.04$. This seemingly linear increase of the costs with n is not the entire truth since it ignores the drop of the acceptance for the lowest n and does not take autocorrelation into account. As a matter of fact the integrated autocorrelation time itself is a noisy quantity and our statistics by far not sufficient to allow for a reliable estimate. Looking e.g. at the autocorrelation times for the average plaquette or the PCAC mass m_1,[1] we find for both observables $\tau_{\text{int}} \approx 4 \pm 2$ in units of

[1]The definition is $m_1 = \left[(\partial_0^* + \partial_0) f_A(x_0)/2 + c_A a \partial_0^* \partial_0 f_P(x_0)\right] / (2 f_P(x_0))$.

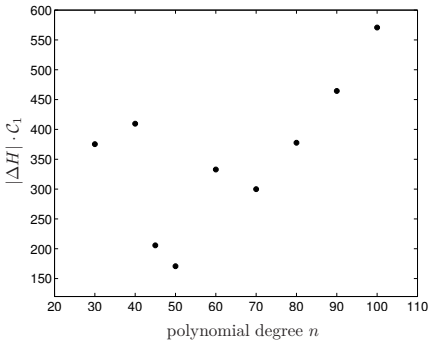

Figure 8.2. Estimate of the costs including the performance of the algorithm.

MD-time without any pattern with respect to the degree of the polynomial n. Since the acceptancse reveals due to limited statistics only coarse information, we prefer to multiply $\mathcal{C}_1$ instead by $|\langle \Delta H \rangle|$. This quantity reflects the performance of the algorithm since a high acceptance rate will have a small ΔH, while a period of rejections leads to a large ΔH. To keep statistical fluctuations of $|\Delta H|$ at a minimum we always started for each n from the same initial configuration and the same set of pseudo random numbers. According to Fig. 8.2, $n = 50$ is optimal, whereas the low acceptance occurring for $n = 20$ leads to a huge $\Delta H = 1.2$ and consequently very high costs, which are not shown for that reason.

The next tests keep the degree of the polynomial fixed ($n = 40$) and vary each at a time $\hat{e}$ and $\hat{\delta}$, while the other parameter is at its optimal value. Due to the mapping relation of the hopping-operator to its even-odd preconditioned version (eq. (4.19)) one expects $\hat{e} \approx \hat{\delta}$ up to effects due to the clover term. Hence we show the dependence on both quantities in one plot.

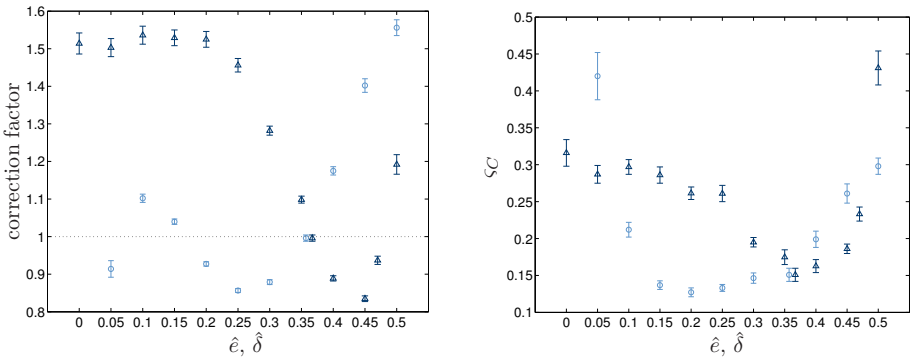

Figure 8.3. Dependence of the correction factor C (left) and its distribution measured by ς_C (right) on $\hat{e}$ (dark blue $\triangle$) and $\hat{\delta}$ (light blue $\circ$).

Looking at the value of the correction factor C (Fig. 8.3, left) we see a strong dependence on $\hat{e}$ and $\hat{\delta}$ with our optimal choices, $\hat{e}_{\text{opt}}$ and $\hat{\delta}_{\text{opt}}$ leading to a value closest to 1. Increasing $\hat{\delta}$ from 0, C meanders around 1 ± 0.15, but grows strongly for $\hat{\delta} > 0.4$. When varying $\hat{e}$, we observe in the range $[0, \ldots, 0.2]$ that the correction factor is around 1.5, drops down to 0.8 at $\hat{e} = 0.45$ before rising again. If we turn to the correction factor's distribution, ς_C, (Fig. 8.3 right) the behavior is different. ς_C favors strongly the optimal choice for $\hat{e}$ (smallest fluctuations), while when varying $\hat{\delta}$ only a weak dependence is seen in the range $[0.15, \ldots, 0.4]$. We find a similar picture by looking at the number of CG iterations (8.4). The number decreases for $\hat{e}$ approaching its optimal value, while changing $\hat{\delta}$ shows only a little effect. All in all, these findings confirm the results obtained in Section 4.3.2. $\hat{e}_{\text{opt}}$ and $\hat{\delta}_{\text{opt}}$ are indeed optimal choices.

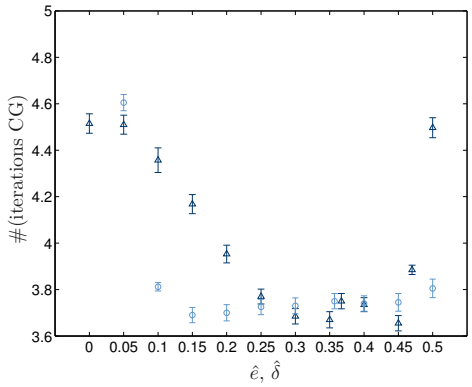

Figure 8.4. Dependence of the number of CG iterations when varying $\hat{e}$ (dark blue $\triangle$) or $\hat{\delta}$ (light blue $\circ$).

8.2 Tuning Parameters for Two Pseudo-Fermions

The Hasenbusch-trick allows to split the original problem in two parts and introduces the additional parameter $0 \leq \varrho \leq 1$. First we aim to find a good choice of ϱ before seeking the optimal degrees of the polynomials or the best choice with respect to the costs.

By factorizing the determinant (6.31) we formally double the original problem since for both factors the determinant of a matrix of the original size has to be computed. Hence only by an appropriate prescription how to compute the factorized expression we can yield a gain. Computing the stochastic estimation of the determinants as part of an HMC-type algorithm

belongs to that as well as the choice of ϱ. The two kernels of the pseudo-fermions can be written as

$$M_1 = \mathbb{1} - \varrho\hat{K} \quad \text{and} \quad M_2 = \left[\mathbb{1} - \varrho\hat{K}\right]^{-1} \hat{M}. \tag{8.1}$$

Looking at the two limits of ϱ, we see that M_1 becomes trivial for $\varrho \to 0$, while M_2 becomes the original problem $\hat{M}$. For $\varrho \to 1$ the situation is similar but the roles of M_1 and M_2 are exchanged. In both cases we are left with one operator corresponding to quark fields showing "little interaction" and one operator describing the "interacting" quark fields. In dependence of ϱ our implementation (see Chapter 6) allows to lower the degree of the polynomial approximating M_1 easily without loosing precision, whereas M_2 requires always a high degree polynomial, which can only be carefully optimized to maintain a good initial guess for the CG.

A good choice for ϱ leads to smaller fermionic forces during the MD evolution, thus smaller energy violations occur resulting in a higher acceptance rate. In this sense we are using the absolute average value of $|\langle\Delta H\rangle|$ to detect a good choice on ϱ. ΔH is an easily available quantity and provides direct information of the performance of the algorithm but has the drawback to provide only cumulated information related to all occurring forces.[2] Again we keep statistical fluctuations of $|\Delta H|$ at a minimum by always starting from the same initial configuration and the same pseudo random numbers. Measuring $|\langle\Delta H\rangle|$ for four different setups we find that both extrema ($\varrho \to 0, 1$) lead to a higher acceptance rate because of smaller energy violations. The simulation parameters are listed in Table 8.1 and we perform 25, in case of the 16^4 lattice 15, trajectories of unit length for each ϱ with single time scale integration scheme. The degree of the polynomial approximating M_2^{-1} is always given by $n_{\max}$, whereas we reduced the degree to approximate M_1^{-1} in dependence of ϱ ensuring its sufficiency by checking that $\varsigma_{C,1}$ is well below 0.15. According to our observation in the previous section, a fine tuning of $\hat{e}$ and $\hat{\delta}$ is not required and values in the range $0.32 \leq \hat{e} = \hat{\delta} \leq 0.36$ are taken.

As can be concluded from Fig. 8.5, good choices for the Hasenbusch parameter are $\varrho \leq 0.25$ and $\varrho \geq 0.99$, if looking e.g. at the 16^4 data. In principal one aims for a ϱ such that both pseudo-fermions have clearly distinct force contribution but none is trivial since then MTS integration becomes most advantageous. If ϱ is too small (say around 0.1), we have the situation that M_1 contributes little to the total force and can be cheaply approximated, while M_2 requires an expensive approximation but is the bulk contribution to the total force. Hence in that case one can only gain from the splitting

[2] For technical reasons the measurement of the forces as explained in the previous section is currently not available.

lattice	$T \times L^3$	β	κ	$\hat{e} = \hat{\delta}$	$\delta\tau$	$n_{\max}$	N_{trj}
$N8_a$	8×8^3	6.3229	0.13577	0.32	0.05	55	25
$N8_b$	8×8^3	6.00	0.13458	0.36	0.0625	50	25
$N8_c$	8×8^3	5.8097	0.13661	0.36	0.05	70	25
$N16$	16×16^3	5.2	0.13568	0.34	0.025	100	15

Table 8.1. Simulation parameters for tuning the NPHMC with two pseudo-fermions.

of the forces but not additionally from MTS integration. This gain can be expected for larger ϱ.

The information collected in Fig. 8.5 is not sufficient for a general statement on good choices for ϱ. Therefore more statistics and a greater variation on lattice sizes is required. However, for the four analyzed setups it seems to be reasonable to test at $\varrho = 0.25$ and 0.9 for gains yielded by MTS integration. In both cases we increase the number of steps performed for the first pseudo-fermion and expect (by fixed step-size of the second pseudo-fermion) a further decrease of $|\langle \Delta H \rangle|$ and an increase of the acceptance.

Selecting two of the four lattices we continue tuning and seek the lowest degree of the polynomials such that $\varsigma_{C,i} < 0.15$ and the number of required CG iterations of the heatbath for the pseudo-fermion or the computation of the correction factor is less than 20. The second constraint arises because otherwise these inversions dominate the cost figures (6.47, 6.48). Since a more demanding problem opens a wider range to tune the n_i we proceed with simulations $N8_c$ and $N16$. Performing 25 trajectories we list the results for different choices of n_i in Tables D.9 and D.10. As expected $|\Delta H|$ is not

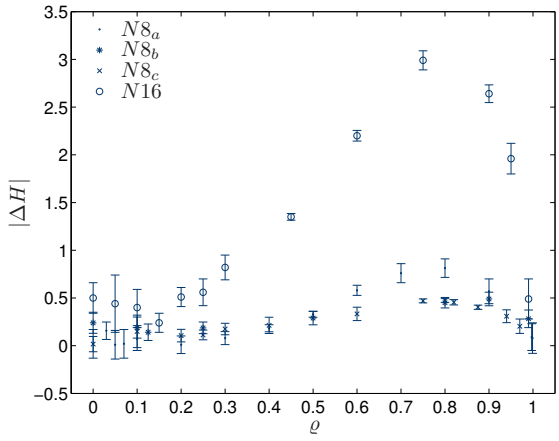

Figure 8.5. Measuring $|\langle \Delta H \rangle|$ in dependence of ϱ for the lattices specified in Table 8.1.

lattice	ϱ	$\delta\tau$	n_1	n_2	s_1/s_2	Acc.	$\lvert\Delta H\rvert$
$N8_c$	0.25	0.05	10	65	1	66%	0.267(74)
					3	88%	0.015(58)
					5	88%	0.022(58)
					7	88%	0.023(58)
					10	88%	0.024(58)
$N8_c$	0.99	0.1	60	55	3	76%	0.051(39)
					5	**96%**	**0.020(13)**
					7	96%	0.014(10)
					10	96%	0.0094(96)
$N16$	0.25	0.025	10	80	3	80%	0.014(25)
					5	80%	0.014(26)
					7	80%	0.014(25)
$N16$	0.99	0.04	75	55	3	87%	0.151(43)
					5	**93%**	**0.027(21)**
					7	93%	0.001(24)

Table 8.2. Seeking optimal values of MTS integration parameters for $\varrho = 0.25$ and 0.99.

affected by the choice of the n_i as long as the n_i do not become too small. Moreover, in case of the second pseudo-fermion we observe, that the quality of the initial guess for the CG is the weak spot. Although its correction factor shows only little fluctuations (tiny $\varsigma_{C,2}$) the number of CG iterations rises quickly reaching prohibitively expensive values. Such a problem is absent for the first pseudo-fermion which behaves much like in case of the one pseudo-fermion algorithm but has a heavier mass.

To explore finally the advantages of MTS integration we fix the degree of the polynomials and vary only the parameter s_1 which specifies how many times the first pseudo-fermion is updated for one step of the second pseudo-fermion. The first pseudo-fermion is updated more frequently since in all cases it is approximated by a polynomial of lower degree and is hence the "cheaper" one. Its effective step size is $\delta\tau \cdot s_2/s_1$.

Table 8.2 confirms that $\varrho = 0.25$ is still too small to yield a gain due to MTS integration. The more frequent integration of M_1 has no effect on the acceptance or the occurring energy violations, unlike for $\varrho = 0.99$. There we could increase the step-size $\delta\tau$ for the $N8_c$ lattice by a factor 2 and find for both lattices with $s_1 = 5$ a raise of the acceptance rate to over 90%.

In order to decide with resepct to the costs which choice of ϱ is better we will use single time scale integration for $\varrho = 0.25$ and MTSI for $\varrho = 0.99$.

8.3 Comparison between the NPHMC and our Standard HMC

We close this chapter by comparing the performance and costs of the new NPHMC algorithm to our standard HMC algorithm. For both we consider the following three variants:

- one pseudo-fermion with Sexton-Weingarten integrator
- two pseudo-fermions with STSI (Sexton-Weingarten integrator)
- two pseudo-fermions with MTSI (Leap-Frog integrator)

Each time we perform 100 trajectories[3] and choose as before the setups corresponding to lattices $N8_c$ and $N16$ (Table 8.1). The idea is to adjust the step size $\delta\tau$ such that roughly equal acceptance rates around 80% are achieved. However the spread turned out to be around 15%.[4] Hence comparing the different variants is more troublesome than expected.

Let us first look at the data from the NPHMC (upper part of Table 8.3). In case of $\varrho = 0.99$ on the 8^4 lattice, the CG iteration number violated for a couple of trajectories the maximal iteration number and hence we decided to enhance the precision of the second pseudo-fermion by increasing the degree of its polynomial approximation from 55 to 70 (compared to our previous studies). Moreover we adjusted the step size $\delta\tau$ on lattices $N16$ for both runs with Sexton-Weingarten integrator. Focussing at the performance we notice that on both lattices MTSI is advantageous: it allows for a factor two larger step size and shows nevertheless significantly higher acceptance than with other integration schemes. STSI integration with two pseudo-fermions shows a similar or even a bit worse performance compared to the one pseudo-fermion algorithm. Hence $\varrho = 0.25$ is not the advantageous choice.

Turning to the data obtained by our standard HMC (lower part of Table 8.3) we find again MTSI to be superior to the other integration schemes.[5] Here one sees a clear order: the one pseudo-fermion algorithm becomes improved by the Hasenbusch-trick and allowing then for multiple time scale integration another gain is realized.

In order to obtain now a comparison among the different variants we require comparable cost figures. Using the cost figures presented in Chapter

[3] In case of the 8^4 lattice we have in addition four replica.

[4] One difficulty arises due to `APE` restrictions to provide the number of steps s_i and the steps size $\delta\tau$ as input parameters but one likes to get *exactly* a trajectory length 1.

[5] The determination of ρ is shown for the 16^4 lattice in Fig. 7.2 and for the 8^4 lattice given by measurements of $|\Delta H|$ as listed in D.11.

lattice	ϱ	$\delta\tau$	$\frac{s_1}{s_2}$	s_2	n_1	n_2	$\#(\mathrm{CG_{HB}})_1$	$\#(\mathrm{CG_{HB}})_2$	$\#(\mathrm{CG_M})$	$\#(\mathrm{CG_{CF}})_1$	$\#(\mathrm{CG_{CF}})_2$	$\lvert\Delta H\rvert$	Acc.
$N8_c$	—	0.05	20	—	70	—	3(0)	—	—	3.135(5)	—	0.167(28)	78%
$N8_c$	0.25	0.05	—	20	10	65	1(0)	3.09(2)	9(0)	1.80(3)	3.2(1)	0.130(20)	81%
$N8_c$	0.99	0.1	5	10	60	70	3(0)	3.7(2)	108.8(5)	2.95(1)	4.1(3)	0.0146(39)	97%
$N16$	—	0.02	50	—	90	—	2.01(1)	—	—	2.33(3)	—	0.040(49)	82%
$N16$	0.25	0.02	—	50	10	80	1(0)	2(0)	8.17(4)	1(0)	1.9(1)	0.268(32)	76%
$N16$	0.99	0.04	5	25	75	55	2.04(3)	3.2(7)	133.9(8)	1.98(4)	3.7(9)	0.0500(69)	95%

lattice	ρ	$\delta\tau$	$\frac{s_1}{s_2}$	s_2	$\#(\mathrm{CG_{HB}})_2$	$\#(\mathrm{CG_{MD}})_1$	$\#(\mathrm{CG_{MD}})_2$	$\lvert\Delta H\rvert$	Acc.
$H8_c$	—	0.05	20	—	—	108.9(8)	—	0.175(33)	77%
$H8_c$	0.05	0.05	—	20	55.85(8)	56.20(7)	124.6(7)	0.065(19)	86%
$H8_c$	0.05	0.1	5	10	55.94(8)	56.21(7)	124.5(6)	0.075(55)	96%
$H16$	—	0.02	50	—	—	138.3(2)	—	0.184(60)	73%
$H16$	0.067	0.03125	—	32	45.91(9)	46.22(5)	164(2)	0.097(74)	79%
$H16$	0.067	0.03125	5	32	45.89(7)	46.25(3)	163(1)	0.011(11)	96%

Table 8.3. Analyzing the performance of the new NPHMC (N) and our standard HMC (H) for lattices $N8_c$ and $N16$ (cf. Table 8.1) by generating 4×100 and 1×100 trajectories, respectively. The CG iteration numbers refer to: HB heatbath of the pseudo-fermion, M inversion of $\hat{M}\hat{M}^\dagger$ when generating the second NPHMC-pseudo-fermion, CF correction factor and MD molecular dynamics. The first line of each set of three corresponds to the one pseudo-fermion, the second to two with STSI and the third to two with MTSI.

6, (6.47) and (6.48), we estimate the numerical costs of the pseudo-fermions when performing one trajectory of a NPHMC run. Similar figures are given for the pseudo-fermions of our standard HMC by

$$C_1' = 2 \cdot \#(\mathrm{CG_{MD}})_1 \cdot s_1 \tag{8.2}$$
$$C_2' = 2 \cdot (\#(\mathrm{CG_{MD}})_2 \cdot s_2 + \#(\mathrm{CG_{HB}})_2). \tag{8.3}$$

Evaluating these cost figures we list the results in Table 8.4 and omit any errors due to the roughness of this method. For the HMC we observe roughly equal costs for both pseudo-fermions in case of MTS integration on the larger lattice, while the NPHMC exhibits always rather different contributions for both pseudo-fermions.

	NPHMC			HMC		
	ϱ	$C_1 + C_2$	Acc.	ρ	$C_1' + C_2'$	Acc.
1 PF	—	3.8	78%	—	4.4	77%
STSI	0.25	4.1	81%	0.05	7.2	86%
MTSI	0.99	9.7	97%	0.05	8.2	96%
1 PF	—	9.9	82%	—	11.0	70%
STSI	0.25	9.9	76%	0.067	13.6	79%
MTSI	0.99	23.1	95%	0.067	25.3	96%

Table 8.4. Estimated costs in units of 10^3 applications of the Dirac-Wilson operator to compute one trajectory of length 1 with the NPHMC or the HMC, respectively. Upper part 8^4 lattice, lower part 16^4 lattice.

Comparing the three variants of the NPHMC with the corresponding ones of the HMC, we have almost the same rate of acceptance and it appears that the NPHMC is slightly superior to the HMC. There is only one exception, the MTS integration on the 8^4 lattice, where the HMC shows a somewhat better performance. If we check the performance of e.g. the NPHMC with one pseudo-fermion versus the one with two and MTSI, a conclusive answer is difficult to give since both acceptance rates differ crucially. Compensating for that by multiplying with $|\Delta H|$ does not offer a solution here since the two algorithm exhibit quite different $|\Delta H|$ although the acceptance rate is almost equal. This answer aims in addition at the question when MTS integration is useful and how much one can gain.

Assuming our cost figures to be a realistic estimate, the costs of MTS integration are a factor 2 larger than the ones occurring e.g. for the one pseudo-fermion algorithm but we gain only little in the acceptance rate. Here arises the suspicion that our MTS integration is too fine spending too many

iterations on the first, "cheaper" pseudo-fermion. Repeating the MTS tests on the 8^4 lattice with only three times integration steps of the first for one step of the second pseudo-fermion we find for the HMC 94% acceptance at costs (in units of 10^3) of approximately 6.0, whereas in case of the NPHMC the acceptance drops to 89% and the costs are around 6.9 [10^3]. This shows that a good tuning of the MTSI parameters is required in order to tune the algorithm to its best performance and be able to decide which one is the best choice. Likely, this answer depends to some extent on the problem to be simulated. The fact that both pseudo-fermions of the NPHMC in case of MTS integration have still rather distinct contributions is certainly a disadvantage limiting a possible gain from MTSI. In the end a better method to set the MTSI parameters is needed and one properly should also take the effect of autocorrelation into account when determining the costs.

Chapter 9

Conclusion and Outlook

In this thesis we study properties of the Dirac-Wilson operator in order to enhance lattice QCD simulations with dynamical fermions. The framework of our analysis are Schrödinger functional boundary conditions for $N_f = 2$ flavor QCD. These studies focus on the one hand on properties of the ($O(a)$ improved) operator itself and, on the other hand, on how one can take advantage of those within a HMC-type algorithm.

Working first with the even-odd preconditioned Hermitian operator, we find that the symmetric version of even-odd preconditioning leads to smaller fermionic forces with a more narrow distribution than the asymmetric version. These advantages allow a larger step size in the numerical integration leading to a gain of up to 30%. The tool to analyze the fermionic forces occurring in a HMC update proves also to be useful when tuning integrator parameters. A common trick to speed-up dynamical fermion simulations is to split the fermion determinant by introducing the Hasenbusch parameter ρ and creating that way two pseudo-fermions. By measuring separately the forces occurring in the update, one can determine the optimal value for ρ and tune a multiple time scale integrator such that expensive parts with a small contribution are computed less often than cheap parts with a larger contribution. Thus an additional gain to the Hasenbusch-trick becomes possible.

Analyzing the stability of HMC simulations employing the Hermitian operator, one is in particular concerned by tiny eigenvalues of the Dirac-Wilson operator since they can cause large energy violations. For a stable run smallest eigenvalues well separated from zero (spectral gap) are therefore desirable. Determining the distribution of the smallest eigenvalues in two different volumes at the same physical parameters we observe that by increasing the volume the distribution's width narrows but also that its median decreases.

A further concern are autocorrelations affecting all Monte Carlo simulations based on a Markov chain. Investigating the dependence of the HMC

algorithm on the length of the trajectory τ we find that longer trajectories ($\tau = 2$) are favored since the autocorrelation is reduced but no significant increase of the reversibility violations are seen. Turning to physical observables, like the pseudo-scalar mass or decay constant, we stress that autocorrelation times vary from observable to observable and have to be computed individually. Evaluating a couple of large volume simulations we finally perform a scaling test in order to investigate cutoff effects on a number of non-perturbatively renormalized quantities.

In order to yield a further speed-up of our two-flavor simulations we decided to study the non-Hermitian Dirac-Wilson-operator and in particular its spectrum with respect to an approximation of its inverse by Chebyshev polynomials. Theses studies lead to the development of the NPHMC algorithm as update for two-flavor lattice QCD simulations and is the central aspect of this thesis. At the heart of the algorithm is the approximation of the inverse non-Hermitian Dirac-Wilson operator by complex, scaled and translated Chebyshev polynomials. These polynomials allow for simple and stable recurrence relations which carry over to a straight forward implementation.

The choice of these polynomials and their parameters are motivated by theoretical and numerical considerations on the spectrum of the non-Hermitian operator. We found new insights on peculiar features of Schrödinger functional boundary conditions, as well as an explanation why the symmetric version of even-odd preconditioning is superior to the asymmetric version. It proved to be useful to estimate the eigenvalues on the spectral boundary by the complex Lanczos method. Besides visualizing the effects of preconditioning or the $O(a)$-improvement by the Sheikholeslami-Wohlert term, we have direct access to two parameters required for the polynomial approximation. Testing the quality of these approximations by monitoring the exponential decay of the remainder, we found promising results.

Based on the same Chebyshev polynomials we develop the NPHMC algorithm which compensates by reweighting for a deviation from importance sampling. The dependence of the algorithm on the various input parameters is analyzed. Moreover, we extend the basic algorithm to incorporate the Hasenbusch-trick allowing a split of the determinant and giving thus rise to two pseudo-fermions.

Judging conclusively the performance of the NPHMC algorithm from the first tests is difficult and work is still ongoing. To exclude effects stemming from e.g. different boundary conditions or linear algebra routines, we restrict ourselves to a comparison of the NPHMC with our standard HMC in the setup of Schrödinger Functional boundary conditions. Comparing both algorithms without the additional feature of the Hasenbusch-trick, the NPHMC

is superior to the HMC since one profits from the well-working polynomial approximation. Here one can easily take advantage of the deviation from importance sampling. With two pseudo-fermions generated by the Hasenbusch-trick the HMC can be tuned such that its performance is better than the one pseudo-fermion NPHMC. Due to the determinant break-up the forces in the MD evolution decrease which can in particular be exploited by MTSI. Adding the Hasenbusch-trick to the NPHMC tuning the various parameters becomes tricky and a clear, general statement for the two pseudo-fermion variant is yet not possible, but a gain as big as for the HMC is unlikely.

For two pseudo-fermions the NPHMC requires the tuning of several parameters which strongly influence the performance and some of which depend on each other. In particular before starting a simulation, the Hasenbusch parameter ϱ and both degrees of the approximating polynomials must be set appropriately in addition to the step size $\delta\tau$. The need for this complicated tuning is a disadvantage of the NPHMC limiting its immediate practicality. Once a good working choice is found the algorithm performs probably smoother than the standard HMC because of two reasons: on the one hand by the polynomial approximation extremal eigenvalues are only taken into account by the reweighting factor (deviation from importance sampling) and on the other hand, the number of steps during the force computation is fixed. This is reflected by the reversibility violations being of the order of machine accuracy. However, there are iterative inversions at the beginning and the end of a trajectory (for the heatbath of the pseudo-fermions and when computing the correction factor) which can become prohibitively expensive for an unfortunate choice of parameters.

These tuning problems may relax with increasing experience of the algorithm's behavior and one may exploit the freedom to use a coarser polynomial for the guidance than for the acceptance Hamiltonian. Also measuring the different forces during the MD evolution will help to tune the parameters and probably allow for more profound statements than by just looking at $|\Delta H|$. With this help it should become easier to simulate / investigate more challenging lattice like the ones used in Chapter 7 and check how the algorithm's performance scales with increasing volume. Furthermore one has to address questions regarding the performance during thermalization – in particular with respect to the problem of finding the optimal parameter set. Next a performance comparison should be done with a cost figure including the autocorrelation time like in [77], i.e. one has to look at the costs to generate sufficiently independent configurations. Here like for other interesting algorithmic properties sufficient statistics is mandatory and a study for its own sake not affordable. Hence one should keep an eye on these properties when using this algorithm within a physical research program.

Extension to 2+1 flavor simulations

The polynomial hybrid Monte Carlo is one candidate for simulating $2+1$ flavors, i.e. two light and degenerate (u- and d-quark) and one heavier (s-quark) flavor are simulated. For sufficiently heavy quark masses it can be safely assumed that $\hat{M}$ has only eigenvalues with positive real part and $\det\{\hat{M}\}$ can be estimated by a bosonic integral. Nevertheless a standard HMC is not possible since the generation of the pseudo-fermion fields ϕ requires to multiply $\sqrt{\hat{M}}$ in order to achieve the appropriate sampling. A solution to this problem is based on a polynomial approximation of the non-Hermitian operator.[33, 96]

Assuming the spectrum of $\hat{M} = \mathbb{1} - \hat{K}$ to be entirely in the right complex half plane, we approximate $\hat{M}^{-1}$ by Chebyshev polynomials using the root factorization. First we obtain a sum which can be rewritten as product of monomials

$$P_N(\hat{M}) = \sum_{i=0}^{N} c_i (\hat{M} - 1)^i = c_N \prod_{j=1}^{N} (\hat{M} - z_j), \tag{9.1}$$

where N is an even degree of the approximating polynomial and the z_j are the complex roots appearing in complex conjugate pairs. Exploiting the latter property we achieve

$$P_N(\hat{M}) = c_N \prod_{k=1}^{N/2} (\hat{M} - z^*_{j'(k)})(\hat{M} - z_{j(k)}). \tag{9.2}$$

$j'(k)$ and $j(k)$ serve for reordering the indices thus $z^*_{j(k)} = z^*_{j'(k)}$ is fulfilled with $j = 1, \ldots, N/2$. By the pseudo-Hermiticity of $\hat{M}$ (eq. (3.6) or (3.12), respectively) one finds that $\det\{\hat{M} - z^*_{k'(j)}\} = \det\{\hat{M} - z_{k(j)}\}^\dagger$ and can write (9.2) as

$$\begin{aligned} \det\{P_N(\hat{M})\} &= c_N \prod_{k=1}^{N/2} \det\{\hat{M} - z_{j(k)}\}^\dagger \det\{\hat{M} - z_{j(k)}\} \\ &= \det\{T_N^\dagger(\hat{M}) T_N(\hat{M})\} \end{aligned} \tag{9.3}$$

with

$$T_N(\hat{M}) \equiv \sqrt{c_N} \prod_{k=1}^{N/2} (\hat{M} - z_{j(k)}). \tag{9.4}$$

Now the pseudo-fermion field ϕ can be generated by applying T_N^{-1}, whereby these polynomials are considered as "square root" of $\hat{M}^{-1}$.

In case of our polynomial approximation we use simple and stable recurrence relations instead of the root factorization. Hence this idea can not be transferred. To follow our concept it appears to be more suitable to seek an approximation of $\hat{M}^{-1/2}$ in terms of different polynomials. As already discussed in [97, 98], a possible starting point are here the *Legendre polynomials*, which are like the Chebyshev polynomials a special case of the *Gegenbauer polynomials*. As before we are forced to scale and translate $\hat{M}$ such that we have an origin centered spectrum enclosed by the smallest ellipse with focal points at ± 1 which determines the two parameters c and t

$$\hat{M} = c(1 + t^2 - 2tA). \tag{9.5}$$

Then an approximation of $\hat{M}^{-1/2}$ in terms of the Legendre polynomials $L_n(A)$ is given by

$$(1 + t^2 - 2tA)^{-1/2} = \sum_{n=0}^{\infty} t^n L_n(A). \tag{9.6}$$

The L_n obey the two-step recurrence relation

$$\begin{aligned} (n+1)L_{n+1}(A) &= (2n+1)AL_n(A) - nL_{n-1}(A) \\ \text{with} \quad L_1 &= A \quad \text{and} \quad L_0 = \mathbb{1}\,. \end{aligned} \tag{9.7}$$

Due to the required summation in (9.6), this approximation looks less appealing than our recursions for the Chebyshev polynomials (6.7) and (6.8). Finding a suitable transcription here is desirable but unfortunately not obvious.

Bibliography

[1] M. E. Peskin and D. V. Schroeder. *Introduction to Quantum Field Theroy*. Westview Press, Boulder, 1995.

[2] W. N. Cottingham and D. A. Greenwood. *An Introduction to the Standard Model of Particle Physics*. Cambridge University Press, Cambridge, 2. edition, 2007.

[3] T. Muta. *Foundations of Quantum Chromodynamics*. World Scientific Publishing, Singapore, 2. edition, 2000.

[4] K. G. Wilson. Confinement of Quarks. *Phys. Rev.*, D10:2445–2459, 1974.

[5] R. Sommer. A new way to set the energy scale in lattice gauge theories and its applications to the static force and α_s in $SU(2)$ Yang-Mills theory. *Nucl. Phys.*, B411:839–854, 1994.

[6] M. Della Morte *et al.* Computation of the strong coupling in QCD with two dynamical flavours. *Nucl. Phys.*, B713:378–406, 2005.

[7] M. Della Morte *et al.* Scaling test of two-flavor $O(a)$-improved lattice QCD. *to appear in JHEP*, 2008.

[8] I. Montvay and G. Münster. *Quantum fields on a lattice*. Cambridge University Press, Cambridge, 1994.

[9] N. Cabibbo and E. Marinari. A New Method for Updating $SU(N)$ Matrices in Computer Simulations of Gauge Theories. *Phys. Lett.*, B119:387–390, 1982.

[10] M. Creutz. Monte Carlo Study of Quantized $SU(2)$ Gauge Theory. *Phys. Rev.*, D21:2308–2315, 1980.

[11] K. Fabricius and O. Haan. Heat bath method for the twisted Eguchi-Kawai model. *Phys. Lett.*, B143:459, 1984.

[12] M. Creutz. Overrelaxation and Monte Carlo Simulation. *Phys. Rev.*, D36:515, 1987.

[13] S. Aoki *et al.* Quenched light hadron spectrum. *Phys. Rev. Lett.*, 84: 238–241, 2000.

[14] S. Aoki *et al.* Light hadron spectrum and quark masses from quenched lattice QCD. *Phys. Rev.*, D67:034503, 2003.

[15] S. Aoki *et al.* Light hadron spectroscopy with two flavors of O(a)-improved dynamical quarks. *Phys. Rev.*, D68:054502, 2003.

[16] T. Ishikawa *et al.* 2+1 flavor light hadron spectrum and quark masses with the $O(a)$ improved Wilson-clover quark formalism. *PoS*, LAT2006: 181, 2006.

[17] Th. DeGrand and C. DeTar. *Lattice Methods for Quantum Chromodynamics*. World Scientific Publishing Co. Pte. Ltd., 2006.

[18] H. B. Nielsen and M. Ninomiya. No Go Theorem for Regularizing Chiral Fermions. *Phys. Lett.*, B105:219, 1981.

[19] K. Symanzik. Continuum Limit and Improved Action in Lattice Theories. 1. Principles and ϕ^4 Theory. *Nucl. Phys.*, B226:187, 1983.

[20] K. Symanzik. Continuum Limit and Improved Action in Lattice Theories. 2. O(N) Nonlinear Sigma Model in Perturbation Theory. *Nucl. Phys.*, B226:205, 1983.

[21] B. Sheikholeslami and R. Wohlert. Improved Continuum Limit Lattice Action for QCD with Wilson Fermions. *Nucl. Phys.*, B259:572, 1985.

[22] M. Lüscher, R. Narayanan, P. Weisz, and U. Wolff. The Schrödinger functional: A Renormalizable probe for non-Abelian gauge theories. *Nucl. Phys.*, B384:168–228, 1992.

[23] St. Sint. On the Schrodinger functional in QCD. *Nucl. Phys.*, B421: 135–158, 1994.

[24] M. Lüscher. Advanced lattice QCD, 1998. hep-lat/9802029.

[25] U. Wolff. Nonhermitian Polynomial Hybrid Monte Carlo. Internal Notes, Jan. 2001.

[26] Th. A. Manteuffel. The Tchebychev Iteration for Nonsymmetric Linear Systems. *Numer. Math.*, 28:307–327, 1977.

[27] A. D. Kennedy. Algorithms for dynamical fermions, 2006. Write-up ILFTN Workshop 'Perspectives in Lattice QCD', Nara, 2005.

[28] B. Bunk. Chebyshev Polynomials, Iterative Solvers and Matrix Inversion. Internal Notes, Sep. 1997.

[29] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, 1993.

[30] W. Gautschi. Computational Aspects of Three-Term Recurrence Relations. *SIAM Review*, 9:24–82, 1967.

[31] P. Deufelhard and A. Hohmann. *Numerische Mathematik I.* de Gruyter, 2002. 3. Auflage 380 S.

[32] K. Jansen and C. Liu. Implementation of Symanzik's improvement program for simulations of dynamical Wilson fermions in lattice QCD. *Comput. Phys. Commun.*, 99:221–234, 1997.

[33] S. Aoki *et al.* Polynomial hybrid Monte Carlo algorithm for lattice QCD with odd number of flavors. *Phys. Rev.*, D65:094507, 2002.

[34] H. B. Meyer and O. Witzel. Symmetric Even-Odd-Preconditioning. Internal Notes, Nov. 2006.

[35] S. Takeda, O. Witzel, and U. Wolff. Spectral properties of the non-hermitian Wilson-Dirac operator in the Schroedinger functional. *PoS*, LAT2007:046, 2007.

[36] G. H. Golub and Ch. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, London, 3. edition, 1996.

[37] B. Bunk. Wilson Fermions. Internal Notes, Sep. 1997.

[38] Herbert Neuberger. Bounds on the Wilson Dirac operator. *Phys. Rev.*, D61:085015, 2000.

[39] H. Neuberger. Adler's Overrelaxation Algorithm for Goldstone Bosons. *Phys. Rev. Lett.*, 59:1877, 1987.

[40] B. Bunk. Preconditioning. Internal Notes, Aug. 1999.

[41] R. Halíř and J. Flusser. Numerically stable direct least squares fitting of ellipses. *Proc. Int. Conf. in Central Europe on Computer Graphics, Visualization and Interactive Digital Media.*, pages 125–132, 1998.

[42] J. Heitger. Scaling tests in $O(a)$-improved quenched lattice QCD. *Nucl. Phys. Proc. Suppl.*, 73:921–923, 1999.

[43] M. Lüscher, St. Sint, R. Sommer, P. Weisz, and U. Wolff. Non-perturbative $O(a)$ improvement of lattice QCD. *Nucl. Phys.*, B491: 323–343, 1997.

[44] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, and A. H. Teller. Equation of State Calculations by Fast Computing Maschines. *The Jour. of Chem. Phys.*, 21:1087–1092, 1953.

[45] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid Monte Carlo. *Phys. Lett.*, B195:216–222, 1987.

[46] D. J. E. Callaway and A. Rahman. Microcanonical ensemble formulation of lattice gauge theory. *Phys. Rev. Lett.*, 49(9):613–616, 1982.

[47] D. J. E. Callaway and A. Rahman. Lattice gauge theory in the microcanonical ensemble. *Phys. Rev. D*, 28(6):1506–1514, 1983.

[48] S. Duane and J. B. Kogut. Hybrid Stochastic Differential Equations Applied to Quantum Chromodynamics. *Phys. Rev. Lett.*, 55:2774, 1985.

[49] S. Duane and J. B. Kogut. The Therory of Hybrid Stochastic Algorithms. *Nucl. Phys.*, B275:398, 1986.

[50] J. C. Sexton and D. H. Weingarten. Hamiltonian evolution for the hybrid Monte Carlo algorithm. *Nucl. Phys.*, B380:665–678, 1992.

[51] I. P. Omelyan, I. M. Mryglod, and R. Folk. Construction of high-order force-gradient algorithms for integration of motion in classical and quantum systems. *Phys. Rev. E*, 66:026701, 2002.

[52] M. Hasenbusch. Speeding up the Hybrid-Monte-Carlo algorithm for dynamical fermions. *Phys. Lett.*, B519:177–182, 2001.

[53] M. Hasenbusch and K. Jansen. Speeding up lattice QCD simulations with clover-improved Wilson fermions. *Nucl. Phys.*, B659:299–320, 2003.

[54] M. Hasenbusch. Speeding up the HMC algorithm: Some new results. *PoS*, LAT2005:116, 2005.

[55] A. Alexandru and A. Hasenfratz. Partial-global stochastic Metropolis update for dynamical smeared link fermions. *Phys. Rev.*, D66:094502, 2002.

[56] M. Lüscher. A New approach to the problem of dynamical quarks in numerical simulations of lattice QCD. *Nucl. Phys.*, B418:637–648, 1994.

[57] Ph. de Forcrand and T. Takaishi. Fast fermion Monte Carlo. *Nucl. Phys. Proc. Suppl.*, 53:968–970, 1997.

[58] R. Frezzotti and K. Jansen. A polynomial hybrid Monte Carlo algorithm. *Phys. Lett.*, B402:328–334, 1997.

[59] R. Frezzotti and K. Jansen. The PHMC algorithm for simulations of dynamical fermions. I: Description and properties. *Nucl. Phys.*, B555: 395–431, 1999.

[60] R. Frezzotti and K. Jansen. The PHMC algorithm for simulations of dynamical fermions. II: Performance analysis. *Nucl. Phys.*, B555: 432–453, 1999.

[61] B. Bunk *et al.* A New simulation algorithm for lattice QCD with dynamical quarks. *Nucl. Phys. Proc. Suppl.*, 42:49–55, 1995.

[62] B. Bunk, St. Elser, R. Frezzotti, and K. Jansen. Ordering monomial factors of polynomials in the product representation. *Comput. Phys. Commun.*, 118:95–109, 1999.

[63] M. Lüscher. Comment on the PHMC algorithm, Dec. 2002. http://luscher.web.cern.ch/luscher/.

[64] T. Takaishi and Ph. de Forcrand. Odd-flavor Simulations by the Hybrid Monte Carlo, 2000. Proceedings of the workshop on Non-perturbative methods and lattice QCD in Guangzhou.

[65] T. Takaishi and Ph. de Forcrand. Odd-flavor hybrid Monte Carlo algorithm for lattice QCD. *Int. J. Mod. Phys.*, C13:343–366, 2002.

[66] A. Borici and Ph. de Forcrand. Systematic errors of Lüscher's fermion method and its extensions. *Nucl. Phys.*, B454:645–662, 1995.

[67] A. Borrelli, Ph. de Forcrand, and A. Galli. Non-hermitian Exact Local Bosonic Algorithm for Dynamical Quarks. *Nucl. Phys.*, B477:809–834, 1996.

[68] I. Montvay and E. E. Scholz. Updating algorithms with multi-step stochastic correction. *Phys. Lett.*, B623:73–79, 2005.

[69] E. E. Scholz and I. Montvay. Multi-step stochastic correction in dynamical fermion updating algorithms. *PoS*, LAT2006:037, 2006.

[70] M. A. Clark and A. D. Kennedy. The RHMC algorithm for 2 flavors of dynamical staggered fermions. *Nucl. Phys. Proc. Suppl.*, 129:850–852, 2004.

[71] M. A. Clark. The rational hybrid Monte Carlo algorithm. *PoS*, LAT2006:004, 2006.

[72] A. Frommer, B. Nockel, St. Gusken, Th. Lippert, and K. Schilling. Many masses on one stroke: Economic computation of quark propagators. *Int. J. Mod. Phys.*, C6:627–638, 1995.

[73] M. A. Clark and A. D. Kennedy. Accelerating fermionic molecular dynamics. *Nucl. Phys. Proc. Suppl.*, 140:838–840, 2005.

[74] M. A. Clark and A. D. Kennedy. Accelerating dynamical fermion computations using the rational hybrid Monte Carlo (RHMC) algorithm with multiple pseudofermion fields. *Phys. Rev. Lett.*, 98:051601, 2007.

[75] M. Lüscher. Lattice QCD and the Schwarz alternating procedure. *JHEP*, 05:052, 2003.

[76] M. Lüscher. Solution of the Dirac equation in lattice QCD using a domain decomposition method. *Comput. Phys. Commun.*, 156:209–220, 2004.

[77] M. Lüscher. Schwarz-preconditioned HMC algorithm for two-flavour lattice QCD. *Comput. Phys. Commun.*, 165:199–220, 2005.

[78] M. Lüscher. Deflation acceleration of lattice QCD simulations. *JHEP*, 12:011, 2007.

[79] St. Sint and R. Sommer. The Running coupling from the QCD Schrodinger functional: A One loop analysis. *Nucl. Phys.*, B465:71–98, 1996.

[80] M. Göckeler *et al.* Determination of light and strange quark masses from full lattice QCD. *Phys. Lett.*, B639:307–311, 2006.

[81] H. B. Meyer *et al.* Exploring the HMC trajectory-length dependence of autocorrelation times in lattice QCD. *Comput. Phys. Commun.*, 176: 91–97, 2007.

[82] H. B. Meyer and O. Witzel. Trajectory length and autocorrelation times: $N(f) = 2$ simulations in the Schroedinger functional. *PoS*, LAT2006:032, 2006.

[83] M. Della Morte *et al.* Preparing for $N_f = 2$ simulations at small lattice spacings. *PoS*, LAT2007:255, 2007.

[84] C. Urbach, K. Jansen, A. Shindler, and U. Wenger. HMC algorithm with multiple time scale integration and mass preconditioning. *Comput. Phys. Commun.*, 174:87–98, 2006.

[85] M. J. Peardon and J. Sexton. Multiple molecular dynamics time-scales in hybrid Monte Carlo fermion simulations. *Nucl. Phys. Proc. Suppl.*, 119:985–987, 2003.

[86] K.-I. Ishikawa *et al.* An application of the UV-filtering preconditioner to the polynomial hybrid Monte Carlo algorithm. *PoS*, LAT2006, 2006.

[87] M. Lüscher, R. Sommer, P. Weisz, and U. Wolff. A Precise determination of the running coupling in the $SU(3)$ Yang-Mills theory. *Nucl. Phys.*, B413:481–502, 1994.

[88] U. Wolff. Monte Carlo errors with less errors. *Comput. Phys. Commun.*, 156:143–153, 2004. Erratum-ibid.176:383,2007.

[89] M. Guagnelli, J. Heitger, R. Sommer, and H. Wittig. Hadron masses and matrix elements from the QCD Schrödinger functional. *Nucl. Phys.*, B560:465–481, 1999.

[90] L. Del Debbio, L. Giusti, M. Lüscher, R. Petronzio, and N. Tantalo. Stability of lattice QCD simulations and the thermodynamic limit. *JHEP*, 02:011, 2006.

[91] Th. Kalkreuter and H. Simma. An Accelerated conjugate gradient algorithm to compute low lying eigenvalues: A Study for the Dirac operator in SU(2) lattice QCD. *Comput. Phys. Commun.*, 93:33–47, 1996.

[92] M. Della Morte, R. Hoffmann, and R. Sommer. Non-perturbative improvement of the axial current for dynamical Wilson fermions. *JHEP*, 03:029, 2005.

[93] L. Del Debbio, L. Giusti, M. Lüscher, R. Petronzio, and N. Tantalo. QCD with light Wilson quarks on fine lattices. II: DD-HMC simulations and data analysis. *JHEP*, 02:082, 2007.

[94] St. Sint and P. Weisz. Further results on $O(a)$ improved lattice QCD to one-loop order of perturbation theory. *Nucl. Phys.*, B502:251–268, 1997.

[95] M. Della Morte *et al.* Non-perturbative quark mass renormalization in two-flavor QCD. *Nucl. Phys.*, B729:117–134, 2005.

[96] C. Alexandrou *et al.* The deconfinement phase transition in one-flavor QCD. *Phys. Rev.*, D60:034504, 1999.

[97] B. Bunk. Fractional Inversion in Krylov Space. *Nucl. Phys. Proc. Suppl.*, B63:952, 1998.

[98] B. Bunk. Gegenbauer Expansions. Internal Notes, Mar. 2004.

[99] I. Schur. Über die charakteristischen Wurzeln einer linearen Substitution mit einer Anwendung auf die Theorie der Integralgleichungen. *Math. Annalen*, 66:488–510, 1909.

[100] O. Toeplitz. Das algebraische Analogon zu einem Satze von Fejér. *Math. Z.*, 2:187–197, 1918.

[101] F. Hausdorff. Der Wertvorrat einer Bilinearform. *Math. Z.*, 3:314–316, 1919.

[102] P. Henrici. Bounds for iterates, inverses, spectral variation and fields of values of non-normal matrices. *Num. Math.*, 4:24–40, 1962.

[103] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, 2. edition, 2000.

[104] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, editors. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide.* SIAM, Philadelphia, 2000.
http://www.cs.utk.edu/∼dongarra/etemplates/book.html.

[105] F. S. Acton. *Numerical Methods That Work*. Harper & Row, New York, 1. edition, 1970.

[106] C. Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Res. Nat. Bur. Standards*, 45:255–282, 1950.

[107] J. K. Cullum and R. A. Willoughby. A QL procedure for computing the eigenvalues of complex symmetric tridiagonal matrices. *SIAM J. Matrix Anal. Appl.*, 17, 1996.

[108] R.B. Lehoucq, D.C. Sorensen, and C. Yang. *ARPACK User's Guide: Solution of Large-Scale Eigenvalue Problems with Implicity Restarted Arnoldi Methods*. SIAM, Philadelphia, 1998.

[109] W.-M. Yao *et al.* Review of Particle Physics. *Journal of Physics G*, 33:1+, 2006. URL `http://pdg.lbl.gov`.

[110] S. Necco and R. Sommer. The $N(f) = 0$ heavy quark potential from short to intermediate distances. *Nucl. Phys.*, B622:328–346, 2002.

[111] M. Della Morte, R. Hoffmann, F. Knechtli, R. Sommer, and U. Wolff. Non-perturbative renormalization of the axial current with dynamical Wilson fermions. *JHEP*, 07:007, 2005.

[112] M. Della Morte, R. Sommer, and S. Takeda. Work in progress, 2008.

Appendix A

Norms and Matrices

In this appendix we provide the definitions of the used matrix norms and discuss furthermore aspects of normal and non-normal matrices. Finally, we outline the conjugate gradient (CG) a commonly used algorithm to numerically invert positive-definite and Hermitian matrices.

A.1 Norms

Suppose φ is a complex vector. Its (Euclidean) 2-norm is defined by

$$\|\varphi\| = \sqrt{\varphi^\dagger \varphi} \tag{A.1}$$

and has the properties

$$\|\varphi\| = 0 \Rightarrow \varphi = 0 \tag{A.2}$$
$$\|\phi_1 + \phi_2\| \leq \|\phi_1\| + \|\phi_2\| \tag{A.3}$$
$$\|c\varphi\| = |c| \cdot \|\varphi\| \quad c \in \mathbb{C}. \tag{A.4}$$

A unit vector with respect to $\|\cdot\|$ satisfies $\|\varphi\| = 1$.

The vector 2-norm induces a matrix 2-norm for a matrix $A \in \mathbb{C}^{n\times n}$

$$\|A\| = \sup_{\varphi\neq 0} \frac{\|A\varphi\|}{\|\varphi\|} = \max_{\|\varphi\|=1} \|A\varphi\| \tag{A.5}$$

thus $\|A\|$ is the 2-norm of the largest vector when applying the matrix A to a unit vector.[36] For the matrix 2-norm holds (in addition to the corresponding properties (A.2) - (A.4))

$$\|A\varphi\| \leq \|A\| \cdot \|\varphi\| \tag{A.6}$$
$$\|A_1 \cdot A_2\| \leq \|A_1\| \cdot \|A_2\|. \tag{A.7}$$

The 2-norm of a general matrix A can be expressed by the spectrum of $A^\dagger A$

$$\|A\| = \sqrt{\max_i \lambda_i(A^\dagger A)}, \tag{A.8}$$

where λ_i denotes the eigenvalues of A. If A is normal, $[A, A^\dagger] = 0$, (A.8) reduces to

$$\|A\| = \max_i |\lambda_i(A)|. \tag{A.9}$$

Besides the matrix 2-norm (spectral norm) we introduce the Frobenius (Euclidean) norm (indicated by a subscript "F") for a $(m \times n)$-matrix A

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} \overset{m=n}{=} \sqrt{\mathrm{Tr}\ (A^\dagger A)} = \sqrt{\sum_{i=1}^{n}\lambda_i\,(A^\dagger A)}. \tag{A.10}$$

If A is a $(n \times n)$-square matrix Frobenius and 2-norm obey the inequality

$$\|A\| \leq \|A\|_F \leq \sqrt{n}\|A\|. \tag{A.11}$$

Since for the 2-norm as well as the Frobenius norm holds

$$\|UAU^\dagger\| = \|A\|, \tag{A.12}$$

where U is a unitary transformation, both are called *unitarily invariant*.

A.2 Normal vs. Non-normal Matrices

A $(n \times n)$-matrix A is defined to be *normal* if and only if

$$A^\dagger A = AA^\dagger, \tag{A.13}$$

while a square matrix violating eq. (A.13) is *non-normal*.

Both normal and non-normal matrices can be transformed to an upper triangular form by terms of an unitary transformation (*Schur decomposition*)[99]

$$U^\dagger AU = S = D + N. \tag{A.14}$$

Here and in the following U denotes an unitary matrix, $UU^\dagger = \mathbb{1}$, and S an upper triangular matrix which has only on and above the diagonal non-zero entries. S can be split into a diagonal matrix D built up by the eigenvalues λ_i of A, i.e. $D = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, and a strictly upper triangular and thus

nil-potent matrix N. In general the Schur decomposition is not unique and an arbitrary order of eigenvalues can be achieved.

If the matrix A is *normal* the Schur decomposition can be extended to the *spectral decomposition*, i.e. N vanishes and A is diagonalizable by an *unitary transform*

$$A = UDU^{\dagger}. \tag{A.15}$$

Then A has a complete set of ortho-normal eigenvectors. Moreover, Toeplitz showed that the eigenvalues of A lie within a convex hull $H(A)$ which matches the fields of values (Wertevorrat) $F(A)$[100]. The field of values is defined to be the set of all complex numbers

$$F(A) = \left\{ \frac{\zeta^{\dagger} A \zeta}{\zeta^{\dagger}\zeta} : 0 \neq \zeta \in \mathbb{C}^n \right\}. \tag{A.16}$$

Computing the r^{th} power of a *normal* matrix A we find that its 2-norm is bounded by the r^{th} power of the spectral radius of A which is A's maximal eigenvalue $\lambda_{\max} = \max_i |\lambda_i(A)|$

$$\|A^r\| = \lambda_{\max}^r. \tag{A.17}$$

For $\lambda_{\max} < 1$, $\|A^r\|$ converges monotonically to zero for $r \to \infty$.

Turning our attention to *non-normal* matrices one finds that the field of values $F(A)$ is still convex but may extend beyond the convex hull of eigenvalues $H(A)$ [101]. Following Henrici we derive a bound on the distance of the boundary of $F(A)$ to the convex hull $H(A)$ [102].

Therefore, we show first that the Frobenius norm of N is unique

$$\begin{aligned} \|A\|_F^2 = \|U(D+N)U^{\dagger}\|_F^2 = \|D+N\|_F^2 = \sum_{i=1}^{n} |\lambda_i|^2 + \sum_{i<j} |n_{ij}|^2 \\ \Rightarrow \|N\|_F = \sqrt{\|A\|_F^2 - \sum_{i=1}^{n} |\lambda_i|^2}, \end{aligned} \tag{A.18}$$

using the fact that the Frobenius norm is unitarily invariant and that the $\sum_{i,j=1}^{n} |d_{ij} + n_{ij}|$ can be split because always one summand vanishes. Thus the Frobenius norm on N can serve as a measure on the non-normality of A.

Next we start from the Schur decomposition (eq. (A.14)) and multiply the vectors $\zeta^{\dagger}$ and ζ from the left and from the right, respectively, whereby $\zeta^{\dagger}\zeta = 1$ which defines f, a point of the field of values

$$f \equiv \zeta^{\dagger} A \zeta = \zeta^{\dagger} U (D+N) U^{\dagger} \zeta. \tag{A.19}$$

Introducing $\eta = U^\dagger \zeta$ we still have $\eta^\dagger \eta = 1$ and denote by $h = \eta^\dagger D \eta$ a point in the convex hull. Hence (A.19) can be turned into

$$|f - h| = |\eta^\dagger N \eta| \tag{A.20}$$

and it remains to estimate the rhs. By the Cauchy inequality we find

$$|\eta^\dagger N \eta|^2 = \left| \sum_{i<j} n_{ij} \eta_i^* \eta_j \right|^2 \leq \|N\|_F^2 \sum_{i<j} |\eta_i \eta_j|^2 \leq \|N\|_F^2 \cdot \frac{1}{2} \left(1 - \frac{1}{n}\right), \tag{A.21}$$

and hence we obtain the desired bound on the distance from the boundary of the field of values to the convex hull

$$|f - h| \leq \sqrt{\frac{1 - n^{-1}}{2}} \|N\|_F, \tag{A.22}$$

which vanishes for *normal* matrices.

Finally, we seek a bound on powers of *non-normal* matrices to generalize (A.17). Like in the case of *normal* matrices the 2-norm is appropriate. Hence we require a measure of the non-normality using the 2-norm

$$\Delta = \inf \|N\|, \tag{A.23}$$

where the infimum has to be taken w.r.t. *all* N that can occur in the Schur decomposition (A.14), which itself is not unique. To obtain a bound on $\|A^r\|$ we have to expand $(D + N)^r$ respecting that D and N do not commute in general. Simplifications arise because D is diagonal and thus any term containing more than $(n-1)$ N's has to vanish. In fact there are $\binom{r}{q}$ terms with q N's and $(r-q)$ D's for $q = 0, 1, \ldots, n-1$. Taking the 2-norm we can replace D by $\lambda_{\max}$ and N by Δ and yield

$$\|A^r\| \leq \lambda_{\max}^r + \binom{r}{1} \lambda_{\max}^{r-1} \Delta + \ldots + \binom{r}{n-1} \lambda_{\max}^{r-n+1} \Delta^{n-1}, \tag{A.24}$$

where we assume a non-vanishing spectral radius, i.e. $\lambda_{\max} > 0$. Again we recover in the limit of vanishing N the result for *normal* matrices.

A.3 Matrix Inversion

The *conjugate gradient* (CG) method is a widely used tool for numerical inversion of large sparse matrices (see e.g. [29, 36, 103]). This algorithm works if the matrix A is positive-definite and Hermitian. In cases of a non-Hermitian matrix B we multiply $B^\dagger$ and compute the inversion of $A = B^\dagger B$

(also known as CG normal equation). As is emphasized in [29] this forces a condition number being the square of the condition number for inverting B only (by using e.g. a variant of the CG called BiCG). However, the CG has the advantage that its convergence is (theoretically) guaranteed.

The idea of the CG is to solve the equation

$$A \cdot x = b, \tag{A.25}$$

where x, b are vectors, by minimizing the function

$$f(x) = \frac{1}{2}(x, A \cdot x) - (x, b). \tag{A.26}$$

The basic algorithm is [103]

$$\begin{aligned}
r_0 =& b - Ax_0 \\
p_0 =& r_0 \\
&\texttt{while } (\|r_j\| > \epsilon) \texttt{ do} \\
&\quad j = j + 1 \\
&\quad \alpha_j = (r_j, r_j)/(p_j, Ap_j) \\
&\quad x_{j+1} = x_j + \alpha_j p_j \\
&\quad r_{j+1} = r_j - \alpha_j A p_j \\
&\quad \beta_j = (r_{j+1}, r_{j+1})/(r_j, r_j) \\
&\quad p_{j+1} = r_{j+1} + \beta_j p_j \\
&\texttt{end}
\end{aligned}$$

The start vector x_0 is either zero or an "initial guess" of the solution to speed up the inversion. The precision is specified by ϵ and commonly one ensures termination by allowing only a maximal number of iterations. If that number is reached the inversion fails.

Appendix B

Non-Hermitian Eigenvalue Problem

To compute the eigenvalues λ_i of any given matrix A one applies similarity transformations in order to yield a diagonal matrix D with the eigenvalues of A.[1] These λ_i solve the eigenvalue equations

$$y_i^\dagger A = \lambda_i y_i^\dagger; \qquad Ax_i = \lambda x_i, \tag{B.1}$$

where y_i are left and x_i right eigenvectors of A.

Assuming A to be a general, non-Hermitian, $(n \times n)$-square matrix (without further properties, like e.g. symmetry ($A = A^T$)) we consider three different kind of similarity transformations. The discussion follows reference [104] and also [29, 36, 103, 105] provide useful information.

- **Unitary transformation.** For any non-Hermitian matrix A there exist a unitary matrix U that transforms A to upper triangular form

  $$S = U^\dagger A U. \tag{B.2}$$

  S is called the Schur form of A and the eigenvalues of A are along the diagonal of S and arbitrary order may be achieved. Although mathematically this transformation looks promising in practice it is of little use since there is in general no practical algorithm that reduces A to the Schur form (for $n > 3$ and excluding trivial cases). Reducing A first to an upper Hessenberg matrix[2] becomes prohibitively expensive since $O(n^3)$ operations are required.

[1] As stated in A.2 non-normal matrices cannot be diagonalized by a unitary transform but diagonal form may be achieved for some matrices using *non-unitary* transformations.

[2] An upper Hessenberg matrix is a square matrix having zeros below the first subdiagonal.

- **Orthogonal transformation.** A non-Hermitian matrix does *not* have an orthogonal[3] set of eigenvectors. Hence

$$D = Q^T A Q, \tag{B.3}$$

with Q orthogonal and D diagonal does not exist. (The Jacobi algorithm applies orthogonal transforms but works for Hermitian matrices only.)

- **Non-orthogonal transformation.** Using a non-orthogonal transformation most non-Hermitian matrices can be transformed to diagonal form

$$D = X^{-1} A X. \tag{B.4}$$

If this transformation is not possible the matrix A is *defective* and does not have a complete set of eigenvectors. Then X becomes singular and can be a source of numerical instability.

Albeit the general non-Hermitian eigenvalue problem may not have a solution, practically, one may not encounter this case. A strategy to avoid X becoming singular is to us orthogonal or close to orthogonal transformations. Moreover only a small fraction of the spectrum may be computed and we do not compute any eigenvectors.

Both methods presented are so called *iterative* methods yielding a partial result after a relatively small number of iterations and are based on the idea of Krylov subspaces.

B.1 Lanczos' Method

By the name "Lanczos' method" we refer to the computation of eigenvalues accomplished in two steps: First the asymmetric, bi-orthogonal Lanczos algorithm [106] is applied to the non-Hermitian matrix A to create a tridiagonal complex-symmetric matrix T which partially represents the spectrum of A. Then in the second step T is diagonalized using a complex version of the QL procedure.[107]

B.1.1 Bi-orthogonal Lanczos Procedure

The asymmetric Lanczos procedure constructs bi-orthogonal Krylov-bases $\mathcal{K}(A, p_1)$ and $\hat{\mathcal{K}}(A^\dagger, q_1)$ which build up the transformation matrices. Thereby

[3] Orthogonal for complex matrices refers to $Q^T Q = \mathbb{1}$.

the Krylov sequences $\{q_1, A^\dagger q_1, A^{\dagger 2} q_1, \ldots\}$ and $\{p_1, Ap_1, A^2 p_1, \ldots\}$ are created which give the Lanczos vectors $\{q_i\}$ and $\{p_i\}$. q_i and p_i are bi-orthogonal, i.e. $p_l^\dagger q_k = \delta_{lk}$ and form the $n \times j$ transformation matrices

$$Q_j = [q_1, q_2, \ldots, q_j] \quad \text{and} \quad P_j = [p_1, p_2, \ldots, p_j], \tag{B.5}$$

which give by multiplication on A the tridiagonal matrix

$$T_j = P_j^\dagger A Q_j = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \gamma_1 & \alpha_2 & \beta_2 & & \\ & \gamma_2 & \alpha_3 & \ddots & \\ & & \ddots & \ddots & \beta_{j-1} \\ & & & \gamma_{j-1} & \alpha_j \end{bmatrix}. \tag{B.6}$$

Instead of performing the expensive matrix multiplication, the α, β and γ are determined recursively

$$\begin{aligned} \beta_i q_{i+1} &= A q_i - \alpha_i q_i - \gamma_{i-1} q_{i-1} \equiv r_i \\ \gamma_i^* p_{i+1} &= A^\dagger p_i - \alpha_i^* p_i - \beta_{i-1}^* p_{i-1} \equiv s_i \end{aligned} \tag{B.7}$$

with

$$\begin{aligned} \alpha_i &= p_i^\dagger A q_i = p_i^\dagger (A q_i - \gamma_{i-1} q_{i-1}) \\ \beta_i \gamma_i &= s_i^\dagger r_i \equiv \omega_i. \end{aligned} \tag{B.8}$$

The described procedure still allows for two choices: First, the starting vectors can be chosen satisfying bi-orthogonality. Here we use $q_1 = p_1$ drawn from a uniform distribution and normalized to 1, thus $p_1^\dagger q_1 = 1$. Secondly only the product $\beta_i \gamma_i \equiv \omega_i$ enters. For later convenience we adopt the convention

$$\beta_i = \gamma_i = \sqrt{\omega_i} \tag{B.9}$$

with a free choice of the sign of the square root to yield right away a symmetric tridiagonal matrix.

The algorithm faces a breakdown if ω_i vanishes but $\|r_i\|, \|s_i\| \neq 0$. Due to limited precision also a near breakdown $\|\omega_i\| \leq \sqrt{\epsilon \|r_i\| \|s_i\|}$ may occur, while $r_i = 0$ or $s_i = 0$ leads to a successful termination. In cases of a (near) breakdown there are strategies how to restart (cf. [104]). Since in our practice no breakdowns are encountered restarts are not implemented.

The bi-orthogonal Lanczos procedure has the inherent problem that due to rounding errors eigenvectors of extremal eigenvalues are re-introduced and not suppressed by an explicit orthogonalization as they are in Arnoldi's method.

B.1.2 QL-Procedure with Implicit Shifts for Tridiagonal Complex-Symmetric Matrices

Once the much smaller tridiagonal matrix is obtained we diagonalize it applying the tridiagonal QL procedure with implicit shifts [29] generalized to complex-symmetric matrices [107]. Following the usual nomenclature the diagonal elements are named d_i and the off-diagonal ones e_i.

This procedure assumes that the complex-symmetric tridiagonal matrix has a QL factorization, where Q is orthogonal and L lower triangular, which is a priori not guaranteed. Hence breakdowns of this procedure are possible, although this rarely happens in practice. The reduction to diagonal form is achieved by a sequence of orthogonal transformations:

$$\begin{aligned} A_s &= Q_s \cdot L_s \\ A_{s+1} &= L_s \cdot Q_s = Q_s^T \cdot A_s \cdot Q_s. \end{aligned} \tag{B.10}$$

To accelerate the convergence A is shifted by a constant k_s thus the decomposition reads

$$\begin{aligned} A_s - k_s \mathbb{1} &= Q_s \cdot L_s \\ A_{s+1} &= L_s \cdot Q_s + k_s \mathbb{1} = Q_s^T \cdot A_s Q_s. \end{aligned} \tag{B.11}$$

Commonly, the Wilkinson shift is used to determine k_s. Computing the eigenvalues of the uppermost nontrivial 2×2 matrix, k_s is set to the value closest to the $(1,1)$ element of that submatrix. Since T is symmetric this determination simplifies to

$$\begin{aligned} &\det \begin{bmatrix} d_l - k_s & e_l \\ e_l & d_{l+1} - k_s \end{bmatrix} = 0 \\ &\Rightarrow (d_l - k_s)(d_{l+1} - k_s) = e_l^2 \\ &\Rightarrow k_s = \frac{d_l + d_{l+1}}{2} \pm \sqrt{\frac{1}{4}(d_l - d_{l+1})^2 + e_l^2}. \end{aligned} \tag{B.12}$$

Introducing $\tilde{g} = (d_{l+1} - d_l)/(2e_l)$ and $\tilde{r} = \sqrt{\tilde{g}^2 + 1}$ the last expression is written as

$$k_s = d_l - \frac{e_l}{\tilde{g} \mp \tilde{r}}, \tag{B.13}$$

where the sign in the denominator is chosen such that $|\tilde{g} \mp \tilde{r}|$ is maximal.

After calculating the Wilkinson shift the diagonalization starts by a plane rotation which is followed by Givens rotations to restore the tridiagonal form. Rotating the (l, l+1) plane corresponds to the following matrix multiplication

$$\left[\begin{array}{c|cc|c} \mathbf{1} & & & \\ \hline & c & -s & \\ & s & c & \\ \hline & & & \mathbf{1} \end{array}\right] \left[\begin{array}{c|cc|c} * & e_{l-1} & 0 & \\ \hline e_{l-1} & d_l & b & f \\ 0 & b & d_{l+1}-p & g \\ \hline & f & g & * \end{array}\right] \left[\begin{array}{c|cc|c} \mathbf{1} & & & \\ \hline & c & s & \\ & -s & c & \\ \hline & & & \mathbf{1} \end{array}\right] = \left[\begin{array}{c|cc|c} * & b' & f' & \\ \hline b' & d_l-p' & g' & 0 \\ f' & g' & d'_{l+1} & e'_{l+1} \\ \hline & 0 & e_{l+1} & * \end{array}\right], \tag{B.14}$$

which are carried out by computing the last matrix explicitly according to

$$\begin{aligned} c &= g/r \\ s &= f/r \quad \text{with } r = \sqrt{f^2+g^2} \\ \Rightarrow e'_{l+1} &= r \qquad\qquad (B.15) \\ d_l - p' &= c^2 d_l - 2csb + s^2(d_{l+1}-p) \\ \Rightarrow p' &= s\left(2cb + s(d_l - d_{l+1} + p)\right) \qquad\qquad (B.16) \\ d'_{l+1} &= s^2 d_l + 2csb + c^2(d_{l+1}-p) = d_{l+1} - p + p' \\ g' &= csd_l + (c^2-s^2)b - cs(d_{l+1}-p) = c\left(2cb + s(d_l - d_{l+1} + p)\right) - b \\ b' &= ce_{l-1} \\ f' &= se_{l-1} \qquad\qquad (B.17) \end{aligned}$$

and deferred to the next iteration follow

$$d'_l = d_l - p' \tag{B.18}$$

$$e'_l = g' \tag{B.19}$$

$$e'_m = 0. \tag{B.20}$$

Note, that by the last equation the wrong assignment (B.15) is overwritten and that for the final rotation (B.18) and (B.19) are dropped. The initial rotation is coded by the initialization

$$\begin{aligned} g &= d_m - k_s \\ s &= c = 1 \\ p &= 0 \\ \Rightarrow f &= e_{m-1} \quad \text{and} \quad b = e_{m-1}. \end{aligned} \tag{B.21}$$

Possible breakdown may occur when computing $c = g/r$ and $s = f/r$ with $r = \sqrt{f^2+g^2}$ in cases where $f^2+g^2 = 0$ i.e. $f = \pm ig$ since for large $|c|$, $|s|$ the algorithm becomes numerically instable. Hence for $|f| \leq |g|$ we define $w = f/g$ with $|w| \leq 1$ and c, s are computed by

$$c = 1/\sqrt{1+w^2}; \qquad s = w/\sqrt{1+w^2}. \tag{B.22}$$

If $w \to \pm i$ a breakdown will still happen and therefore we interrupt the computation if

$$|\sqrt{1+w^2}| < \delta \quad \text{with} \quad \delta = 0.005. \tag{B.23}$$

According to [107] this seems to be safe for matrices generated by the bi-orthogonal Lanczos method. No possible implementation to recover from such a breakdown is coded.

B.2 Arnoldi's Method

We use the term "Arnoldi's method" for the computation of eigenvalues of A which is accomplished by first projecting A orthogonally onto the Krylov subspace and creating a Hessenberg matrix H_m of modest size. Secondly, the eigenvalues of H_m (also known as Ritz eigenvalues) are computed e.g. with a standard QR algorithm.

Arnoldi Procedure

The Arnoldi procedure builds an orthogonal basis of the Krylov subspace using the stabilized Gram-Schmidt process. The created sequence of orthonormal vectors $q_1, q_2, \ldots, q_m$ span the Krylov subspace $\mathcal{K}(A, q_1)$ and are called *Arnoldi vectors*.

Starting with a random vector normalized to one, q_1, the procedure is as follows [104]

```
for j = 1, 2, ..., m do
    w = A q_j
    for i = 1, 2, ..., j do
        h_ij = w^† q_i
        w = w − h_ij q_i
    end
    h_{j+1,j} = ‖w‖
    if h_{j+1,j} == 0, stop
    q_j + 1 = w / h_{j+1,j}
end
```

The orthogonality is constructed in the inner loop, where q_j is projected in the directions of $q_1, \ldots, q_{j-1}$. A breakdown happens if q_j is zero since then the starting vector q_1 is a combination of j eigenvectors.

From the algorithm one derives the relation

$$Aq_j = \sum_{i=1}^{j+1} h_{ij} q_i, \quad \text{for } j = 1, 2, \ldots m. \tag{B.24}$$

Denoting by Q_m the $(n \times m)$-matrix build from the Arnoldi vectors $q_1, \ldots, q_m$ and the $(m \times m)$ Hessenberg matrix by H_m we can turn eq. (B.24) into

$$AQ_m = Q_m H_m + h_{m+1,m} q_{m+1} e_m^\dagger \tag{B.25}$$

which becomes after multiplying both sides with $Q_m^\dagger$ and exploiting the orthogonality

$$Q_m^\dagger A Q_m = H_m. \tag{B.26}$$

Moreover Q matrices of subsequent iterations obey the relation

$$AQ_m = Q_{m+1}\tilde{H}_m \quad \text{with} \quad \tilde{H}_m = \begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} & \ldots & h_{1,m} \\ h_{2,1} & h_{2,2} & h_{2,3} & \ldots & h_{2,m} \\ 0 & h_{3,2} & h_{3,3} & \ldots & h_{3,m} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & 0 & h_{m,m-1} & h_{m,m} \\ 0 & 0 & \ldots & 0 & h_{m,m+1} \end{bmatrix} \tag{B.27}$$

Computing the eigenvalues of H_m one typically finds that these Ritz values converge to the extreme eigenvalues of A. The details of this behavior are not fully understood yet.[104]

Obviously, this method becomes memory and time consuming when computing a larger number of eigenvalues m. This difficulty gets circumvented by *restarting* the algorithm. If one is e.g. interested in eigenvalues of *largest real part* the approximate eigenvector of the eigenvalue with largest real part obtained after a run with m Arnoldi vectors is computed and fed as initial vector for the next run of the Arnoldi method. Due to restarts extremal eigenvalues of the spectrum can be obtained. A detailed description of the widely used implementation `ARPACK` using implicit restarts can be found in [108].

Appendix C

Statistical Analysis

The described tests and Monte Carlos Simulations are analyzed using methods of the statistical error analysis which we like to summarize here to introduce our notation. Let us first assume our data are independent, before considering how to deal with (auto)correlated data.

C.1 Uncorrelated Data

One set of data (1 replicum)

Suppose we performed N measurements of a quantity x with underlying Gaussian distribution. The (arithmetic) *mean value* $\bar{x}$ is then defined to be the quantity minimizing the square of all deviations

$$S = \sum_{i=1}^{N} (\bar{x} - x_i)^2 = \min, \tag{C.1}$$

which is achieved if $\mathrm{d}S/\mathrm{d}\bar{x}$ vanishes. Hence one obtains the well known relation

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i. \tag{C.2}$$

In the limit of infinite many measurements the mean value approaches the *true value* X

$$X = \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} x_i, \tag{C.3}$$

which itself is a non-accessible quantity. Defining the *absolute error* as difference to the true value we have for each measurement an error $e_i = X - x_i$

and the mean value $\bar{x}$ has the error $\bar{e} = X - \bar{x}$, which can be expressed as

$$\bar{e} = X - \bar{x} \overset{(C.2)}{=} X - \frac{1}{N}\sum_{i=1}^{N} x_i = \frac{1}{N}\sum_{i=1}^{N} e_i. \tag{C.4}$$

The last equation states that the error of the mean value is the average of the errors e_i, the errors of each measurement. Squaring eq. (C.4) we obtain

$$\begin{aligned}\bar{e}^2 &= \frac{1}{N^2}\left(\sum_{i=1}^{N} e_i^2\right) = \frac{1}{N^2}\sum_{i=1}^{N} e_i^2 + \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j\neq i} e_i e_j \\ &\approx \frac{1}{N^2}\sum_{i=1}^{N} e_i^2,\end{aligned} \tag{C.5}$$

where the double sum is neglected because when taking the limit of $n \to \infty$ we find for the second equation in (C.4)

$$\lim_{N\to\infty}\frac{1}{N}\sum_{i=1}^{N} e_i = X - \lim_{N\to\infty}\frac{1}{N}\sum_{i=1}^{N} x_i \overset{(C.3)}{=} X - X = 0, \tag{C.6}$$

and by assumption the errors e_i and e_j are uncorrelated.
By convention

$$\sigma = \sqrt{\overline{e^2}} = \sqrt{\frac{\sum_{i=1}^{N}(X - x_i)^2}{N}} \tag{C.7}$$

is named *standard deviation*, the average deviation of a single measurement, and σ^2 is called *variance*, while

$$\sigma_m = \sqrt{\overline{\bar{e}^2}} = \frac{1}{N}\sqrt{\overline{\sum_{i=1}^{N}(X - x_i)^2}} = \frac{\sigma}{\sqrt{N}} \tag{C.8}$$

is called the *error of the mean value*.

As mentioned above X is in general not available and in consequence we do not know e_i or $\bar{e}$ to determine the relevant quantities σ and σ_m. Therefore we introduce (instead of e_i) f_i, the deviation of each measurement from the mean value $\bar{x}$, and relate it to e_i and $\bar{e}$

$$f_i = \bar{x} - x_i = X - x_i - (X - \bar{x}) = e_i - \bar{e}. \tag{C.9}$$

The *average squared deviation of the mean value* is then given by

$$\begin{aligned}s^2 &= \frac{1}{N}\sum_{i=1}^{N} f_i^2 = \frac{1}{N}\sum_{i=1}^{N}(e_i - \bar{e})^2 \overset{(C.4)}{=} \frac{1}{N}\sum_{i=1}^{N} e_i^2 - \bar{e}^2 \\ &= \sigma^2 - \sigma_m^2 = \sigma^2\left(1 - \frac{1}{N}\right).\end{aligned} \tag{C.10}$$

Rearranging (C.10) we find the sought relations without any reference to X

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(\bar{x} - x_i)^2}{N-1}} \quad \text{and} \quad \sigma_m = \sqrt{\frac{\sum_{i=1}^{N}(\bar{x} - x_i)^2}{N(N-1)}}. \tag{C.11}$$

K set of data (K replica)

If we repeat our N measurements K times we yield K replica each having its own mean value, standard deviation, etc. Therefore, we add another index $j = 1, 2, \ldots, K$ to all quantities running over the replica. Analyzing the total set of data one possibility is to compute the average of all mean values [109]

$$\hat{x} = \frac{1}{K}\sum_{j=1}^{K}\bar{x}_j \tag{C.12}$$

and yield the variance by

$$\hat{\sigma}^2 = \frac{1}{K-1}\sum_{j=1}^{K}(\bar{x}_j - \hat{x})^2. \tag{C.13}$$

The error of the mean value follows then from the relation $\hat{\sigma}_m^2 = \hat{\sigma}^2/K$. Furthermore we may improve our estimators employing the variances σ_j^2 as weights and compute the *weighted average* [109]

$$\begin{aligned} \check{x} &= \frac{1}{w}\sum_{j=1}^{K}\frac{\bar{x}_j}{\sigma_j^2} \quad \text{with} \quad w = \sum_{j=1}^{K}\left(\sigma_j^2\right)^{-1} \\ \check{\sigma}_m^2 &= \frac{1}{w} \quad \text{and} \quad \check{\sigma}^2 = K\check{\sigma}_m^2 = \frac{K}{w}, \end{aligned} \tag{C.14}$$

which has a smaller variance than the unweighted average.

Error Propagation

Dealing with derived quantities, $f = f(x_i)$, their error is computed following the Gaussian law of error propagation

$$\sigma_f = \sqrt{\sum_i \sigma_{x_i}^2 \left(\frac{\partial f}{\partial x_i}\right)^2}. \tag{C.15}$$

C.2 Analyzing Autocorrelated Data

Data generated along a Markov chain can show strong autocorrelation in Monte Carlo time which has to be incorporated to get a reliable error estimate for measured quantities. Commonly a sequence of consecutive measurements is therefore collected in bins and a standard error analysis of the binned data is performed (jackknife analysis).

As pointed out in Ref. [88], this error estimate can be improved by numerically integrating the autocorrelation function for primary or derived observables. Following that reference here, we assume a Markov chain produced (after equilibration) a set of N (primary) measurements x_i which have the *true value* X and the statistical mean value

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i. \tag{C.16}$$

The error of the mean and to the true value is given by $\bar{e} = X - \bar{x}$ and the Monte Carlo estimate of the mean squared deviation from X is

$$\sigma(N,X)^2 = \left\langle \left(\frac{1}{N}\sum_{i=1}^{N}(x_i - X) \right)^2 \right\rangle. \tag{C.17}$$

The latter quantity is now related to the autocorrelation function Γ_X by

$$\sigma(N,X)^2 = \frac{1}{N^2}\sum_{i=1}^{N}\Gamma_X(i-j) \tag{C.18}$$

with

$$\Gamma_X(i-j) = \langle (x_i - X)(x_j - X)\rangle = \Gamma_X(j-i), \tag{C.19}$$

which only depends on the distance between measurement i and j. For $i = j$ we recover the variance

$$\Gamma_X(0) = \left\langle (x_i - X)^2 \right\rangle = \mathrm{var}(X) \tag{C.20}$$

and get for *independent* measurements $\sigma(N,X)^2 = \mathrm{var}(X)/N$ since

$$\Gamma_X(i-j) = \Gamma_X(0)\delta_{ij}. \tag{C.21}$$

For a typical Markov chain the measurements are *not independent* and Γ_X is normally positive, falling off asymptotically as the distance grows

$$\Gamma_X(t) \propto \exp\{-t/\tau\} \qquad \text{for } t \to \infty, \tag{C.22}$$

and we require for a reliable error estimate $N \gg \tau$. In that case holds approximately for the autocorrelation function

$$\sum_{i,j=1}^{N} \frac{\Gamma_X(i-j)}{\Gamma_X(0)} \approx N \sum_{t=-\infty}^{+\infty} \frac{\Gamma_X(t)}{\Gamma_X(0)} = 2N\tau_{\mathrm{int},X}, \tag{C.23}$$

where we call $\tau_{\mathrm{int},X}$ the integrated autocorrelation time of X. Finally, we obtain the error estimate

$$\sigma(N,X)^2 = 2\tau_{\mathrm{int},X} \frac{\mathrm{var}(X)}{N}. \tag{C.24}$$

(C.24) has the interpretation that the autocorrelation effectively lowers the number of measurements from N to $N/(2\tau_{\mathrm{int},X})$.

In case of *derived quantities* F there is a functional dependence on the primary observables X_α with mean values $\bar{x}_\alpha$ and we assume

$$F \equiv f(X_\alpha), \tag{C.25}$$

where the index α runs over the primary observables. To estimate F we consider $\bar{F} = f(\bar{x}_\alpha)$ and expand it in a Taylor series around X_α

$$\bar{F} = f(X_\alpha) + \sum_\alpha \frac{\partial f}{\partial X_\alpha} \bar{e}_\alpha + \frac{1}{2} \sum_{\alpha,\beta} \frac{\partial f}{\partial X_\alpha} \frac{\partial f}{\partial X_\beta} \bar{e}_\alpha \bar{e}_\beta + \ldots \tag{C.26}$$

Seeking next an expression for the mean squared deviation we get

$$\begin{aligned} \sigma(N,F)^2 &= \left\langle \left(\bar{F} - f(X_\alpha \right)^2 \right\rangle \\ &\simeq \frac{1}{N} \sum_{\alpha,\beta} \frac{\partial f}{\partial X_\alpha} \frac{\partial f}{\partial X_\beta} \sum_{t=-\infty}^{+\infty} \Gamma_{\alpha\beta}(t). \end{aligned} \tag{C.27}$$

As before, for zero (autocorrelation) time separation we recover the expression of the variance

$$\mathrm{var}(F) = \sum_{\alpha,\beta} \frac{\partial f}{\partial X_\alpha} \frac{\partial f}{\partial X_\beta} \Gamma_{\alpha\beta}(0) \tag{C.28}$$

and define the integrated autocorrelation time by

$$2\tau_{\mathrm{int,F}} = \frac{\sum_{t=-\infty}^{+\infty} \sum_{\alpha,\beta} \frac{\partial f}{\partial X_\alpha} \frac{\partial f}{\partial X_\beta} \Gamma_{\alpha\beta}(t)}{\sum_{\alpha,\beta} \frac{\partial f}{\partial X_\alpha} \frac{\partial f}{\partial X_\beta} \Gamma_{\alpha\beta}(0)}. \tag{C.29}$$

Hence the error estimate for F is given by

$$\sigma(N,F)^2 = 2\tau_{\mathrm{int,F}} \frac{\mathrm{var}(F)}{N}. \tag{C.30}$$

In practice, we do not know the *true values* and have to replace X_α by $\bar{x}_\alpha$. Then the autocorrelation function $\Gamma_{\alpha\beta}$ is approximated by

$$\Gamma_{\alpha\beta} = \frac{1}{N}\sum_{i=1}^{N}(x_{\alpha,i} - \bar{x}_\alpha)(x_{\beta,i} - \bar{x}_\beta) \tag{C.31}$$

and similarly the derivatives are taken (numerically) with respect to $\bar{x}_\alpha$. Moreover, summing the autocorrelation function over the entire range of data is not advantages. After the drop-off noise is collected which vanishes only on average. Hence the summation should be truncated appropriately at $W < N$. This method allows further to quantify the error of the error. The final formulae are given by

$$\begin{aligned}
\sigma(N,F)^2 &= \frac{1}{N}\sum_{\alpha\beta}\frac{\partial f}{\partial X_\alpha}\frac{\partial f}{\partial X_\beta}\left[\Gamma_{\alpha\beta}(0) + 2\sum_{t=1}^{W}\Gamma_{\alpha\beta}(t)\right] \\
\sigma(\sigma(N,F)^2)^2 &\approx \sigma(N,F)^2\,\frac{W+0.5}{N} \\
\sigma(\tau_{\mathrm{int,F}})^2 &\approx 4\tau_{\mathrm{int,F}}^2\,\frac{W-\tau_{\mathrm{int,F}}+0.5}{N}.
\end{aligned} \tag{C.32}$$

When looking at the autocorrelation function of an observable it is advantageous to consider the normalized function $\rho(t) = \Gamma(t)/\Gamma(0)$.

Appendix D

Listing of Simulation Parameters and Results

Here we list parameters and results of two-flavor QCD simulations and runs to explore the dependence of the autocorrelation time on the trajectory length τ. In Tables D.1 – D.3 the subscripted integer in brackets denotes the summation window in MD time to estimate τ_{int} of an observable.

D.1 Autocorrelation HMC Simulations

	β	$\frac{a}{r_0}$	τ	acc.	$\frac{\langle \Delta H^2 \rangle}{(L\delta\tau)^4}$	$\langle \lvert \Delta H \rvert_{\leftrightarrow} \rangle$	$\tau_{\text{int}}\left(\frac{\mathrm{d}S}{\mathrm{d}\eta}\right)$	$\tau_{\text{int}}\left(\frac{\mathrm{d}S}{\mathrm{d}\eta}\right)_{[25]}$
8^4	6.086	0.16	$\frac{1}{2}$	97%	0.789(6)	$2 \cdot 10^{-4}$	$6.1(4)_{[64]}$	5.0(2)
			1	97%	0.953(6)	$3 \cdot 10^{-4}$	$5.9(3)_{[36]}$	4.2(1)
			2	97%	1.221(8)	$4 \cdot 10^{-4}$	$3.1(1)_{[36]}$	2.75(7)
			4	96%	1.71(1)	$5 \cdot 10^{-4}$	$3.9(1)_{[44]}$	3.46(9)
12^4	6.364	0.11	$\frac{1}{2}$	94%	0.941(6)	$5 \cdot 10^{-4}$	$9(1)_{[46]}$	5.8(3)
			1	93%	1.134(8)	$6 \cdot 10^{-4}$	$6.4(5)_{[63]}$	4.7(2)
			2	93%	1.40(1)	$9 \cdot 10^{-4}$	$3.3(2)_{[34]}$	3.1(1)
			4	91%	1.92(3)	$12 \cdot 10^{-4}$	$4.4(2)_{[44]}$	3.8(2)
16^4	6.570	0.08	$\frac{1}{2}$	89%	1.08(1)	$9 \cdot 10^{-4}$	$7(1)_{[52]}$	5.3(5)
			1	88%	1.31(2)	$12 \cdot 10^{-4}$	$5.8(6)_{[52]}$	4.2(5)
			2	86%	1.58(2)	$16 \cdot 10^{-4}$	$4.2(4)_{[40]}$	3.5(2)
			4	84%	2.10(4)	$21 \cdot 10^{-4}$	$6.0(4)_{[40]}$	4.8(3)

Table D.1. Pure gauge theory. Runs performed in 32-bit precision, $\delta\tau = 1/30$ and boundary field as defined by "point A" in [87]. a/r_0 is taken from [110].

τ	acc.	$\frac{\langle \Delta H^2 \rangle}{L^3 T \delta \tau^4}$	$\langle \lvert \Delta H \rvert_{\leftrightarrow} \rangle$	$\tau_{\rm int}(f_1)$	$\tau_{\rm int}(f_P)$	$\tau_{\rm int}(m_{PS}^{\rm eff})$	$\tau_{\rm int}(f_{PS}^{\rm eff})$
$\frac{1}{2}$	98%	0.827(5)	$5 \cdot 10^{-4}$	$40(5)_{[270]}$	$75(20)_{[245]}$	25(4)	33(6)
2	98%	1.32(1)	$10 \cdot 10^{-4}$	$24(4)_{[184]}$	$44(8)_{[316]}$	14(2)	19(3)
4	97%	1.87(3)	$11 \cdot 10^{-4}$	$20(3)_{[172]}$	$32(4)_{[256]}$	12(1)	15(2)

Table D.2. Quenched QCD. $8^3 \times 32$ lattice at $\beta = 6.0$ and $\kappa = 0.1338$. Runs performed in 32-bit precision with $\delta\tau = 1/50$. $a/r_0 = 0.19$ is taken from [110].

τ	acc.	$\frac{\langle \Delta H^2 \rangle}{L^3 T \delta \tau^4}$	$\langle \lvert \Delta H \rvert_{\leftrightarrow} \rangle$	$\tau_{\rm int}(f_1)$	$\tau_{\rm int}(f_P)$	$\tau_{\rm int}(m_{PS}^{\rm eff})$	$\tau_{\rm int}(f_{PS}^{\rm eff})$
$\frac{1}{2}$	90%	0.147(4)	$1 \cdot 10^{-4}$	$30(15)_{[145]}$	$45(20)_{[185]}$	9(3)	14(4)
				$23(5)_{[50]}$	$28(7)_{[50]}$		
2	91%	0.164(6)	$3 \cdot 10^{-4}$	$24(8)_{[134]}$	$26(10)_{[134]}$	7(2)	4.2(8)
				$15(3)_{[50]}$	$18(4)_{[50]}$		

Table D.3. Two-flavor QCD. $24^3 \times 32$ lattice at $\beta = 5.3$ and $\kappa = 0.1355$. Runs performed in 64-bit precision with $\delta\tau = 1/32$. $a/r_0 = 0.16$ is taken from [80], $aM_{PS} = 0.325(10)$. These runs match B_1 and B_1' in Section D.2.

D.2 Large Volume Simulations

	β	$(L/a)^3 \times T/a$	κ	L^*/a	Z_A	Z_P
A_1	5.5	$32^3 \times 42$	0.13630	10.68(15)	0.805(5)	0.5008(70)
B_1, B_1'			0.13550			
B_2	5.3	$24^3 \times 32$	0.13590	7.82(6)	0.781(8)	0.4939(34)
B_3			0.13605			
B_4			0.13625			
C_1	5.2	$16^3 \times 32$	0.13568	6.51(12)	0.769(12)	0.4788(5)
C_2		$24^3 \times 32$	0.13568			

Table D.4. Reference scale L^* defined by $\bar{g}^2(L^*) = 5.5$. $Z_{\rm A}$ [111, 112] and $Z_{\rm P}$ [95] are given at scale $\mu_{\rm ren}$.

	mol. dyn.	$N_{\mathrm{rep}} \cdot \tau_{\mathrm{tot}}$	ρ	$\langle N_{\mathrm{CG}}^{(0)} \rangle$	$\langle N_{\mathrm{CG}}^{(1)} \rangle$	P_{acc}
A_1	[LF; 2; 5; 50]	$1 \cdot 4340$	0.019803	170	824	88%
B_1	[SW; 2; 1; 64]	$2 \cdot 2400$	0.0300	100	482	91%
B_1'	[SW; $\frac{1}{2}$; 1; 16]	$2 \cdot 1750$	0.0300	100	485	90%
B_2	[SW; $\frac{1}{2}$; 1; 16]	$2 \cdot 1900$	0.0300	102	729	90%
B_3	[LF; 2; 5; 50]	$2 \cdot 2600$	0.019803	143	905	91%
B_4	[LF; 2; 5; 50]	$2 \cdot 1448$	0.0180	155	1195	87%
C_1	[LF; 2; 5; 64]	$1 \cdot 6500$	0.0198	179	791	96%
C_2	[LF; 2; 5; 80]	$2 \cdot 2080$	0.0198	184	1086	94%

Table D.5. Algorithmic parameters of the simulations. The molecular dynamics is characterized by [Integrator; τ; $\delta\tau_1/\delta\tau_0$; $\tau/\delta\tau_1$], where the integrator can be 'leap-frog' or 'Sexton-Weingarten' and superscripts refer to the two pseudofermions in use. For the gauge force, the SW integrator with $\delta\tau_0/\delta\tau_{\mathrm{g}} = 4$ is used in all cases, and $\langle N_{\mathrm{CG}}^{(k)} \rangle$ is the number of conjugate-gradient iterations used to solve the symmetrically even-odd preconditioned Dirac equation during the trajectory.

$\tau_{\mathrm{int}}[O]$	P	$m\left(\frac{T}{2}\right)$	$\mathrm{m}_{\mathrm{eff}}^{\mathrm{A}}\left(\frac{T}{2}\right)$	$\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}\left(\frac{T}{2}\right)$	$\mathrm{F}_{\mathrm{eff}}\left(\frac{T}{2}\right)$	$\mathrm{m}_{\mathrm{eff}}^{\mathrm{V}}\left(\frac{T}{2}\right)$	$\mathrm{G}_{\mathrm{eff}}\left(\frac{T}{2}\right)$
A_1	5.0(9)	4.9(9)	11(3)	21(6)	10(2)	40(10)	23(7)
B_1	13(3)	5.5(9)	7(1)	16(4)	4.2(7)	23(7)	11(3)
B_1'	6(1)	6(1)	10(2)	22(7)	14(4)	24(8)	12(3)
B_2	4.1(7)	4.1(7)	10(3)	14(4)	8(2)	23(7)	24(8)
B_3	9(2)	3.9(6)	4.7(7)	11(2)	6(1)	11(3)	11(2)
B_4	8(2)	5(1)	6(1)	7(2)	4.6(9)	15(5)	8(2)
C_1	9(2)	5.3(8)	5.2(8)	5.1(8)	4.7(7)	4.9(7)	5.6(9)
C_2	11(3)	6(1)	6(1)	7(1)	3.9(6)	6(1)	6(1)

Table D.6. The integrated autocorrelation times for the plaquette, the current quark mass, the effective pseudo-scalar mass and decay constant, and the effective vector mass. The unit is molecular dynamics time, i.e. trajectories times the length of the trajectory.

	$a\,m$	$a\ \mathrm{m}_{\mathrm{eff}}^{\mathrm{A}}$	$a\,\mathrm{m}_{\mathrm{eff}}^{\mathrm{P}}$	$a\ \mathrm{m}_{\mathrm{eff}}^{\mathrm{V}}$	$\frac{a\ \mathrm{F}_{\mathrm{eff}}}{Z_A\,(1+b_{\mathrm{A}}am_{\mathrm{q}})}$	$\frac{a^2\ \mathrm{G}_{\mathrm{eff}}}{Z_P\,(1+b_{\mathrm{P}}am_{\mathrm{q}})}$
A_1	0.015519(37)	0.1800(20)	0.1793(15)	0.2821(50)	0.05999(42)	0.0629(10)
B_1	0.03388(12)	0.3272(18)	0.3236(16)	0.4520(35)	0.09451(41)	0.1507(14)
B_2	0.019599(95)	0.2391(35)	0.2406(19)	0.3953(51)	0.08442(68)	0.1267(22)
B_3	0.01460(11)	0.2118(24)	0.2066(17)	0.3647(35)	0.07714(60)	0.1170(13)
B_4	0.00727(14)	0.1423(55)	0.1528(20)	0.3058(69)	0.0698(11)	0.0985(15)
C_1	0.01401(21)	0.2173(55)	0.2338(24)	0.4354(60)	0.0877(13)	0.1637(25)
C_2	0.01442(14)	0.2328(39)	0.2261(15)	0.4152(42)	0.08773(67)	0.1614(15)
C_{I}	0.01431(19)	0.2286(97)	0.2282(63)	0.410(14)	0.08772(61)	0.1620(17)

Table D.7. Simulation results for the effective quantities evaluated at $x_0 = T/2$. The bare current quark mass has been averaged over $T/3 \leq x_0 \leq 2T/3$. The last line gives the interpolation of $C_1,\ C_2$, including the corrections described in the text.

	$a\,m$	$a\ \mathrm{M}_{\mathrm{PS}}$	$a\,\mathrm{M}_{\mathrm{V}}$	$\frac{a\ \mathrm{F}_{\mathrm{PS}}}{Z_A\,(1+b_{\mathrm{A}}am_{\mathrm{q}})}$	$\frac{a^2\ \mathrm{G}_{\mathrm{PS}}}{Z_P\,(1+b_{\mathrm{P}}am_{\mathrm{q}})}$
D_1	0.03386(11)	0.3286(10)	0.464(3)	0.0949(13)	0.1512(20)
D_2	0.01957(07)	0.2461(09)	0.401(3)	0.0815(10)	0.1260(16)
D_4	0.00761(07)	0.1499(15)	0.344(9)	0.0689(13)	0.1017(24)

Table D.8. Observables from fits of [93] i.e. $x_0, T \to \infty$. Input parameters β, κ and L/a match those of lattices B_1, B_2, B_4; note that D_4 has been renamed here compared to [93].

D.3 Tuning Polynomial Degrees for Two-Pseudo-Fermion NPHMC

ϱ	$\lvert\Delta H\rvert$	n_1	$\varsigma_{C,1}$	#CG$_1$	n_2	$\varsigma_{C,2}$	#CG$_2$
0.25	0.280(66)	10	0.000262	1.867(77)	70	0.04582(62)	3.47(31)
0.25	**0.208(61)**	**10**	**0.000204**	**1.867(77)**	**65**	**0.0909(41)**	**3.80(28)**
0.25	0.344(62)	5	0.343(87)	3.0	65	0.1059(48)	3.53(34)
0.25	0.288(40)	10	0.000017	1.933(65)	60	—	$\lightning$
0.99	0.16(15)	60	0.1150(42)	2.67(21)	70	0.00930(13)	5.5(12)
0.99	0.17(15)	55	0.141(10)	2.93(23)	70	0.00931(13)	5.5(12)
0.99	0.17(17)	50	0.170(19)	3.53(16)	70	0.00927(12)	5.3(13)
0.99	0.17(15)	60	0.1152(42)	2.67(21)	65	0.01870(29)	9.1(20)
0.99	**0.08(11)**	**60**	**0.0641(28)**	**2.53(22)**	**55**	**0.0171409**	**11.9(19)**
0.99	0.08(11)	60	0.0643(28)	2.53(22)	50	—	$\lightning$

Table D.9. Seeking the lowest n_i such that $\varsigma_{C,i} < 0.15$ and the number of CG iterations for the correction factors are less than 20 on the 8^4 lattice, $N8_c$. $\lightning$ indicates the maximal number of CG iterations (20) is exceeded; bold printed lines the selected degrees for further tests. Only relevant statistical errors are shown.

ϱ	$\lvert\Delta H\rvert$	n_1	$\varsigma_{C,1}$	#CG$_1$	n_2	$\varsigma_{C,2}$	#CG$_2$
0.25	0.56(14)	15	0.0	1.0	80	0.1087(29)	1.80(50)
0.25	**0.56(15)**	**10**	**0.005412(72)**	**1.0**	**80**	**0.1087(29)**	**1.80(50)**
0.25	0.72(17)	10	0.000314	1.47(13)	75	—	$\lightning$
0.99	0.46(14)	90	0.011710(19)	1.560(71)	90	0.0	0.0
0.99	0.48(14)	85	0.03304(52)	1.003(61)	80	0.0	1.0
0.99	0.55(15)	80	0.0528(11)	1.600(65)	70	0.0	0.0
0.99	**0.47(14)**	**75**	**0.0655(14)**	**1.840(88)**	**55**	**0.03385(56)**	**2.36(58)**
0.99	0.68(14)	65	0.1163(53)	2.12(21)	50	0.1355(98)	6.5(11)
0.99	0.66(14)	60	0.242(16)	1.96(17)	40	0.357(26)	11.3(20)

Table D.10. Results for tuning n_i on lattice $N16$. $\lightning$ indicates more than 30 CG iterations are required.

D.4 Tuning ρ for Two-Pseudo-Fermion HMC

ρ	Acc.	$\lvert\Delta H\rvert$	$\#(\mathrm{CG}_{\mathrm{HB}})_1$	$\#(\mathrm{CG}_{\mathrm{MD}})_1$	$\#(\mathrm{CG}_{\mathrm{MD}})_2$
1.00	68%	0.223(44)	—	106.0	—
0.70	56%	0.47(15)	12.2	13.4	108.3
0.40	68%	0.108(96)	13.4	14.0	115.6
0.25	72%	0.250(60)	17.0	17.3	116.6
0.20	84%	0.036(78)	20.0	20.0	114.6
0.15	84%	0.114(53)	24.8	24.9	119.0
0.10	84%	0.005(72)	34.0	34.1	121.2
0.05	92%	0.000(47)	56.0	56.3	124.7
0.01	88%	0.07(12)	102.7	106.6	130.5

Table D.11. Seeking the optimal ρ for the HMC algorithm at $\beta = 5.8097$ and $\kappa = 0.13661$ on a 8^4 lattice by computing 25 trajectories with STSI and $\delta\tau = 0.05$.

Selbständigkeitserklärung

Hiermit erkläre ich, die vorliegende Arbeit selbständig und ohne unerlaubte fremde Hilfe angefertigt zu haben. Andere als die angegebende Literatur und Hilfsmittel wurden nicht benutzt.

Berlin, 07. Juli 2008

Oliver Witzel