Towards Higher Precision Lattice QCD Results: Improved Scale Setting and Domain Decomposition Solvers

DISSERTATION

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

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

von
Ben Straßberger

Präsidentin der Humboldt-Universität zu Berlin:

Prof. Dr. Julia von Blumenthal

Dekanin der Mathematisch-Naturwissenschaftlichen Fakultät:

Prof. Dr. Caren Tischendorf

Gutachter:

1. *Prof. Dr. Rainer Sommer*
2. *Dr. Oliver Bär*
3. *Dr. Michael Peardon*

Betreuer:

Dr. Stefan Schaefer

Tag der mündlichen Prüfung: 9. 2. 2023

Abstract

Lattice QCD simulations strive for higher precision. Here, we study two critical points in the generation of high precision lattice results.

In the first part, we calibrate the lattice spacings of QCD simulation with $2 + 1$ flavors of dynamical fermions. We incorporate new measurements and use additional models for the chiral and continuum extrapolations to refine the result obtained in 2017 [1].

The second part focuses on simulation algorithms. We test an algorithm which promises faster solution of the Dirac equation. We analyze the application of the Finite Element Tear and Interconnect (FETI) algorithm in the context of lattice QCD simulations and compare it to other state-of-the-art domain decomposition solvers. We examine various preconditioners and their effects on the convergence of the solution.

Zusammenfassung

Gitter QCD strebt nach höherer Präzision. Hier untersuchen wir zwei kritische Punkte, die zur Genauigkeit von Gitter-Ergebnissen beitragen.

Im ersten Teil kalibrieren wir Gitterabstände von QCD Simulationen mit $2 + 1$ Arten (flavor) dynamischer Quarks. Dabei nutzen wir neue Messungen und eine mehrere Modelle für den chiralen- und Kontinuumslimes, um die Ergebnisse der 2017 durchgeführten Studie [1] zu verbessern.

Der zweite Teil befasst sich mit Simulationsalgorithmen. Wir testen einen Algorithmus, der eine schnellere Lösung der Dirac-Gleichung verspricht. Wir analysieren die Anwendung des FETI-Algorithmus (Finite Element Tear and Interconnect) im Zusammenhang mit Gitter-QCD-Simulationen und vergleichen ihn mit anderen modernen Lösungsverfahren aus der Klasse der Domänendekompositionslösern. Wir untersuchen verschiedene Präkonditionierer und ihre Auswirkungen auf die Konvergenz der Lösung.

Contents

Introduction **1**

1 Lattice QCD **4**

1.1 Gauge Action 6

1.2 Wilson Fermions 6

1.2.1 Pseudofermion Method 8

I Scale Setting **11**

2 Introduction **12**

3 Simulation **16**

3.1 Ensembles 16

3.2 Reweighting 19

3.3 Observables 20

3.3.1 Gradient Flow Scale 20

3.3.2 Correlators 21

3.3.3 Pseudoscalar Mass 21

3.3.4 PCAC Mass 23

3.3.5 Pseudoscalar Decay Constant 23

3.4 Mass Derivatives of Observables 24

3.5 Finite Volume Effects 25

3.6 Error Analysis and Autocorrelations 26

4 Measurements **27**

4.1 Flow Scale 27

4.2 Correlators 31

4.3 Measurement of Pseudoscalar Observables: Plateau and Fit Ranges 32

4.3.1 Pseudoscalar Mass 32

4.3.2 PCAC Mass 36

4.3.3 Pseudoscalar Decay Constant 37

5 Scale Setting **38**

5.1 Dimensionless Observables 39

5.2 Chiral Trajectory and Mistuning 39

5.2.1 Direction of the Derivative 42

5.2.2 Modeling the Derivatives 43

5.3 Extrapolating Lattice Measurements . . . 45
5.3.1 Chiral Extrapolation . . . 45
5.3.2 Continuum Extrapolation . . . 47
5.3.3 Combined Extrapolation Results . . . 47
5.4 Finding the Physical Point . . . 50
5.5 Determination of the Scale and its Uncertainties . . . 53
5.6 Lattice Spacing . . . 56

6 Conclusion **58**

II FETI Algorithm **61**

7 Introduction **62**

8 Linear Algebra Solvers **63**
8.1 Krylov Subspace Solver . . . 64
8.1.1 Steepest Descend and Conjugate Gradient . . . 64
8.1.2 Minimal Residual . . . 66
8.1.3 Generalized Conjugate Residue . . . 66
8.2 Preconditioning . . . 68
8.2.1 Preconditioned GCR . . . 69
8.3 Deflation . . . 69

9 Domain Decomposition Solvers **72**
9.1 Schwarz Alternating Procedure . . . 73
9.1.1 Deflated Schwarz Alternating Procedure . . . 74
9.1.2 Deflated SAP as a Preconditioner . . . 75
9.2 Finite Element Tear Interconnect Algorithm . . . 76
9.2.1 Lattice Geometry . . . 76
9.2.2 Continuity Conditions . . . 78
9.2.3 An Explicit Example . . . 80
9.2.4 Block Quantities . . . 81
9.2.5 FETI System . . . 83
9.2.6 Preconditioners for the FETI Algorithm . . . 85
9.2.7 Improvements to the FETI Algorithm . . . 86

10 FETI in Lattice QCD **88**
10.1 Setup . . . 89
10.1.1 Problem Sizes . . . 89
10.1.2 Solvers for the FETI Algorithm . . . 91
10.2 The λ System . . . 92
10.2.1 FETI as Direct Solver . . . 92
10.2.2 FETI as Preconditioner . . . 94
10.3 The Block System . . . 98
10.3.1 Explicit Solvers . . . 98
10.3.2 Comparing Iterative Solvers . . . 99
10.3.3 Weights in the Dirac Operator . . . 101
10.3.4 Diagonal Preconditioners . . . 101
10.3.5 Conditioning of the Block System . . . 102

10.3.6 Deflation of the Block System 103

11 Conclusion **106**

A Appendix **109**

Introduction

The standard model of particle physics combines three of the four fundamental forces in one theoretical framework. It describes the electromagnetic interaction as well as the so called strong and weak forces. The only interaction not included in the standard model is gravitation, which is orders of magnitude weaker for all but the most extreme cases. The standard model consists of three generations of quark pairs, three generations of leptons and their corresponding neutrinos, four gauge bosons that act as force carriers and the Higgs boson. The model depends on 19 parameters among which are the six quark and three lepton masses and three gauge coupling strengths. The intrinsic properties of standard model particles are determined by symmetries of the underlying theory.

Complex interactions between these elementary and composite particles are observed in collider experiments. To further the understanding of these interactions, experimental as well as theoretical advancements are needed. While experiments rely on measurements, theoretical research works to develop aspects of the interaction from first principles of the underlying theory. These aspects include the structure and parameters of composite particles and coupling strengths of the involved interactions. If we derive these quantities from theory they can be used as inputs rather than results of subsequent experiments. The goal of theoretical as well as experimental studies is to test the standard model, demonstrate its limits and look for physics beyond the standard model.

In the standard model, the particles are described by fields ϕ in the four dimensional space-time. Particle physics phenomena can be expressed in terms of correlation functions or n-point functions of these fields. Commonly, the n-point functions can not be calculated analytically and need to be approximated. This is the case for the standard model.

Observables are calculated in the *path integral* formalism. After the Wick rotation to imaginary times the exponent in the integral usually[1] becomes real. The integral is now called the Euclidean path integral.

$$\langle O\rangle = Z^{-1}\int \mathcal{D}\left[\phi\right] e^{-S(\phi)}O(\phi), \quad Z = \int \mathcal{D}\left[\phi\right] e^{-S(\phi)} \tag{1}$$

The integral of the desired operator is taken over all involved fields ϕ at all points in space-time. The operator is weighted by the Boltzmann weight e^{-S} defined in terms of the action $S(\phi)$ of the theory. The integral over all fields at all points in space-time is not properly defined. It can be expanded in powers of the couplings and defined and regularized for each term. Alternatively, the integral can be defined by restricting the theory to a lattice that is introduced below. These two approaches separate the phenomena described by the standard model into two categories: perturbative and non-perturbative phenomena.

In *perturbation theory* (PT) the path integral is expanded in powers of the coupling constants. Observables are now expressed as an asymptotic series in powers of the couplings. Clearly

[1] The exponent in the Euclidean path integral is real for QCD without the θ-term.

such an approximation only holds if the couplings are sufficiently small. This is given for the electromagnetic and weak interaction, where perturbation theory is very powerful. A multitude of particles and their parameters have been successfully described by perturbation theory and tested by experimental results. Arguably most impressively, the theory prediction for the anomalous magnetic moment of the electron, calculated using PT, matches the experimental value with a precision of 13 orders of magnitude [2].

QCD exhibits a phenomenon called *running coupling*. The QCD coupling varies for different energies. For high energies the coupling is small and hence the interaction between quarks is weak. This characteristic is called *asymptotic freedom* and was discovered in 1973 [3]. In this case the theory can be solved perturbatively. For small energies, on the other hand, the coupling becomes large and the interaction strong. This leads to the *confinement* of quarks [4, 5] in composite structures called hadrons. At low energies, quarks can combine in triplets to create color neutral Baryons. The most prominent Baryons are the proton and the neutron. They can also combine in a quark anti-quark bound state to form Mesons such as the Pion or the Kaon.

To study the QCD phenomena and strongly interacting objects, a non-perturbative approach is required. The only ab-initio approach known to date is lattice QCD. The fundamental idea is to solve the theory approximately by numerical evaluation of the path integral shown in eq. (1). In order to numerically evaluate the path integral the theory is modified. The spacetime is discretized, meaning a four dimensional regular square lattice is defined. Fermion fields are then restricted to exist only on the lattice sites. Gluon fields are described by gauge fields defined on the links connecting neighboring lattice sites. The lattice regularizes the theory and reduces the degrees of freedom of the path integral. The integral over the spacetime is now transformed into a finite sum over lattice points. The path integral can now be statistically sampled using a *Monte Carlo* process [6–8]. In this way, observables like the mass of the proton that can be expressed in terms of the fields ϕ and correlation functions of the fields are evaluated from first principles.

Lattice QCD methods have been successfully used to study non-perturbative phenomena such as the quark masses [12, 15, 29–31], hadron masses, decay constants and form factors [32–37], CKM matrix elements [38–40], the strong coupling constant [41–43], and many more. An overview of recent lattice QCD results is given in [9]. Currently, the most precise determinations of the strong coupling constant α_s and the fundamental energy scale of QCD Λ_{QCD} stem from lattice QCD simulations [9, 14, 42].

The precision of lattice QCD measurements is steadily increasing. Figure 1 shows the development of the relative errors of the strange quark mass over the past decade. The values are averaged over a number of lattice QCD determinations. 2021: [9, 18, 25–27], 2019: [18, 25–28], 2016: [10–15], 2013: [11, 12, 16, 17], 2011: [11, 12, 14, 19–24]. Over the last decade we have seen a fivefold decrease in the uncertainty.

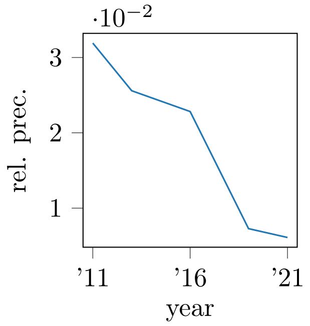


Figure 1: Development of the precision of lattice QCD measurements over the last decade. We show the uncertainty of the averaged measurements of the strange quark mass [9–28]

The QCD contributions to the anomalous magnetic moment of the muon are currently an active field of research in the lattice community [44–48] because of possible discrepancies between theory and experiment [49–51]. Here, high precision results from lattice QCD simulations are essential in increasing the overall accuracy of the theoretical prediction. Additionally, questions about the structure of the proton [52] and other parton distribution functions [53, 54] are still challenging in the lattice QCD framework.

Further advancements in the lattice QCD methods and algorithms are needed to make progress in these fields.

In an effort to further increase the precision of lattice QCD results we consider two aspects of lattice simulations that are critical for the precision of the measurements.

Scale Setting

One key parameter of lattice theory is the lattice spacing a, i.e. the distance between two neighboring lattice sites. On the lattice all dimensionful quantities are compared to the lattice spacing. Consequently, the values of these measurements are only known in physical units once the lattice spacing is known. The process of determining the lattice spacing and thus the reference scale of the simulation is called *setting the scale*. This is a critical step for the precision of lattice QCD simulations as all lattice observables need to be converted to physical units using the reference scale. During the conversion the lattice observables inherit the uncertainty of the scale. For this reason it is advantageous to calculate this scale to high precision. A suitable choice of the scale setting quantity, the gradient flow scale t_0, has been established in previous analyses [1, 55–58]. The gradient flow scale t_0, defined later, is an artificial quantity that can not be measured but calculated on the lattice. In part I of this thesis, we focus on the precise determination of the scale. We update the determination from 2017 in [1], since new measurements and ensembles have become available. In contrast to the previous analysis, we combine measurements from different groups [59–63]. These measurements include ensembles with finer lattice spacings than previously used and one ensemble at the physical point. We also incorporate new measurements of the quark mass derivatives that are needed to shift the measurements to the correct chiral trajectory. Additionally, we compare several extrapolation techniques. With these improvements we are able to increase the precision of the scale by about 20% compared to the determination in [1]. We also boost the confidence in the value of the scale and its uncertainty.

Solution of the Dirac System

Lattice QCD simulations are often limited by the computational resources that are available. The simulation of ever finer lattices and more critical ensembles challenges even the most powerful computers. The most computationally intensive part of the simulation is the solution of the fermion system given by the Dirac operator. The Dirac operator is a large, sparse linear operator whose dimension regularly exceeds 10^7 to 10^9 making it challenging to handle even on the most advanced machines. The problem is further exacerbated by the large condition number of the system. Even with current state-of-the-art preconditioned *domain decomposition* algorithms [64, 65] lattice QCD simulations are often limited by the inversion of the Dirac operator. Any improvements in the simulation algorithms will not only improve the statistics that we are able to produce in a given time, but also extend the feasibility of new simulations. For this reason, in part II of this thesis, we consider the inversion of the fermion system. The Finite Element Tear and Interconnect (FETI) algorithm has shown success in two- and three-dimensional engineering applications [66–68]. It is part of a class of domain decomposition solvers that offer improved scalability compared to global solvers as the size of the system grows. In this work, we test its applicability to the four-dimensional fermion system. We investigate the FETI algorithm as a direct solver and as a preconditioner to a global solver. While we were able to compete with current implementations of the Schwarz Alternating Procedure (SAP), we could not successfully incorporate deflation techniques [69, 70] that have proven instrumental for efficient solver algorithms.

1 Lattice QCD

QCD is the theory of the strong interaction. It describes the interaction between six flavors of quarks mediated by force carriers called gluons. An overview over Quantum Chromodynamics and in particular their lattice formulation of found in [71]. The interactions are governed by the following continuum Lagrangian.

$$\mathcal{L}_{\mathrm{QCD}} = \sum_{f=1}^{N_f} \overline{\psi}^f(x) \left(\gamma_\mu \left(\partial_\mu + iA_\mu(x)\right) + m^f\right) \psi^f(x) + \frac{1}{2g^2} \mathrm{tr}\left(F_{\mu\nu}(x) F_{\mu\nu}(x)\right) \tag{1.1}$$

The Grassmann valued variables $\overline{\psi}^f(x), \psi^f(x)$ describe the different flavors of quarks. Gluons are represented by the gluon field $A_\mu(x)$ and the field strength tensor

$$F_{\mu\nu}(x) = \partial_\mu A_\nu(x) - \partial_\nu A_\mu(x) + i\left[A_\mu(x), A_\nu(x)\right]. \tag{1.2}$$

The Lagrangian and the Euclidean action $S = \int d^4x \mathcal{L}$ separate into the fermionic and gluonic components.

$$S_{\mathrm{QCD}} = S_{\mathrm{F}} + S_{\mathrm{G}} \tag{1.3}$$

$$S_{\mathrm{F}} = \int d^4x \sum_{f=1}^{N_f} \overline{\psi}^f(x) \left(\gamma_\mu \left(\partial_\mu + iA_\mu(x)\right) + m^f\right) \psi^f(x) \tag{1.4}$$

$$S_{\mathrm{G}} = \int d^4x \frac{1}{2g^2} \mathrm{tr}\left(F_{\mu\nu}(x) F_{\mu\nu}(x)\right) \tag{1.5}$$

We are interested in the expectation values $\langle X \rangle$ of a function X of the spinor and gluon fields. In quantum field theories such expectation values are calculated using the Euclidean path integral formalism.

$$\langle X \rangle = \frac{1}{Z} \int \mathcal{D}[A] \mathcal{D}[\psi] \mathcal{D}\left[\overline{\psi}\right] e^{-S_{\mathrm{QCD}}[A,\psi,\overline{\psi}]} X\left[A, \psi, \overline{\psi}\right] \tag{1.6}$$

The integral spans over all gluon and fermion fields.

In perturbation theory the path integral is expanded in powers of the coupling. Because of the running coupling of the strong interaction there are regimes where this expansion in powers of the coupling is not applicable.

Lattice QCD is a perturbation free approach and serves as a way to evaluate the theory in the strong coupling regime. In lattice QCD the spacetime is discretized. A four-dimensional, hypercubic lattice is defined on which the field variables reside. Quarks described by the fermion fields $\overline{\psi}$ and ψ reside on the lattice sites x. Gluons conveying the interaction between quarks

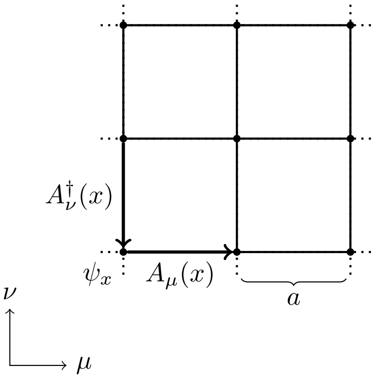


Figure 1.1: Schematic of the lattice geometry in the (μ, ν) plane. Fermion fields $\psi(x)$ reside on the lattice sites, gluon fields $A_\mu(x)$ are defined on the links connecting neighboring sites. The lattice is extended in all directions and continued periodically.

are defined on the links connecting the lattice points. They are indicated by the gluon fields $A_\mu(x)$. The index μ indicates the direction of the link. A two-dimensional cut of the lattice is shown schematically in fig. 1.1. The interaction of quark fields on the lattice sites is limited to neighboring sites along the links. The variable $x = (x_0, x_1, x_2, x_3)^T$ labels the lattice sites, while $\mu = 0, 1, 2, 3$ and $\nu = 0, 1, 2, 3$ indicate the directions on the lattice. The temporal and spacial dimensions are restricted such that $0 \leq x_0 < aT$ and $0 \leq x_{1,2,3} < aL$. The volume of the lattice is $V = a^4 T \times L^3$. At the boundaries the lattice is either continued (anti)periodically or subjected to open [72, 73] or Schrödinger functional [74–77] boundary conditions.

The transformation of the continuum theory to a lattice theory serves several purposes. The lattice spacing a limits the shortest distances and thus acts as a UV regulator. The finite extent of the lattice defines an infrared cutoff. The finite number of lattice points allows us to simulate the theory.

To fully transform the continuum theory onto the lattice we need to take the following steps:

1. Introduce discretized lattice fields.
2. Express the actions in terms of these fields.
3. Translate the continuum functions into lattice functions.
4. Evaluate expectation values of operators on field configurations weighted by the Boltzmann factor $\exp(-S_{\text{lat}})$.

In the first step we restrict the fields to the lattice sites and links respectively. For the lattice representation of the interaction it will be necessary to define the gauge fields $U_\mu(x) \in \text{SU}(3)$

$$U_\mu(x) = \exp\left(iaA_\mu(x)\right). \tag{1.7}$$

The spinor fields are represented by Grassmann-valued variables $\overline{\psi}(x)$ and $\psi(x)$ that reside on the lattice sites.

In step 2 and 3 we express the actions and other operators in terms of these lattice fields. Derivatives in the operators are replaced by their finite difference counterparts

$$\partial_\mu \psi(x) = \frac{\psi(x + a\hat{\mu}) - \psi(x)}{a} + \mathcal{O}(a), \tag{1.8}$$

where $\hat{\mu}$ indicated a unit vector in the direction μ.

It is important to note that the correspondence between the lattice and continuum actions is not unique. A multitude of lattice actions can result in the same continuum action once the continuum limit is taken. The choice of lattice actions and their continuum limits are presented in sections 1.1 and 1.2.

Finally, in item 4, we evaluate the path integral from eq. (1.6). The integral is sampled on a finite number of configurations of the fields. These configurations are generated using a Markov process [78]. Using the set of configurations sampled according to the Boltzmann distribution $e^{-S_{\mathrm{lat}}}$ we approximate the expectation value by

$$\langle X \rangle = \frac{1}{N_{\mathrm{cnfg}}} \sum_{i}^{N_{\mathrm{cnfg}}} X_i \tag{1.9}$$

1.1 Gauge Action

A common formulation of the gauge action that determines the gluonic components of the theory is the Wilson gauge action [3]. It is defined by the sum of all plaquettes on all lattice points x.

$$S_{\mathrm{G}} = \frac{1}{g_0^2} \sum_{x} \sum_{\mu,\nu} \mathrm{tr}\left(1 - U_P^{\mu,\nu}(x)\right) \tag{1.10}$$

The orientation of the plaquette is indicated by the directions μ, ν (see fig. 1.1). The plaquette is calculated from the gauge links according to

$$\begin{aligned} U_P^{\mu,\nu}(x) &= U_\mu(x) U_\nu(x + a\hat{\mu}) U_{-\mu}(x + a\hat{\mu} + a\hat{\nu}) U_{-\nu}(x + a\hat{\nu}) \\ &= U_\mu(x) U_\nu(x + a\hat{\mu}) U_\mu^\dagger(x + a\hat{\nu}) U_\nu^\dagger(x). \end{aligned} \tag{1.11}$$

If we insert the definition of the gauge link from eq. (1.7) and apply the Baker-Campbell-Hausdorff formula we get a relation between the plaquette $U_P^{\mu,\nu}(x)$ and the field strength tensor $F_{\mu\nu}(x)$.

$$U_P^{\mu,\nu}(x) = \exp\left(a^2 F_{\mu\nu}(x) + \mathcal{O}\left(a^3\right)\right) \tag{1.12}$$

$$= 1 + a^2 F_{\mu\nu}(x) + \frac{a^4}{2} F_{\mu\nu}^2(x) + \mathcal{O}\left(a^5\right) \tag{1.13}$$

Inserting and realizing that the field strength tensor is traceless we arrive at

$$S_{\mathrm{G}} = \frac{a^4}{2g_0^2} \sum_{x} \sum_{\mu,\nu} \mathrm{tr}\left(F_{\mu\nu}^2(x)\right) + \mathcal{O}\left(a^2\right). \tag{1.14}$$

If we take the continuum limit $a^4 \sum_x \to \int d^4x$ we arrive at the continuum action in eq. (1.5).

1.2 Wilson Fermions

A lattice formulation for fermions is introduced in [3]. The action for Wilson fermions is derived from the discretization of the derivatives in eq. (1.4). It is defined as

$$S_{\mathrm{W}} = a^4 \sum_{x,f} \overline{\psi}^f(x) \frac{1}{2}\left[\gamma_\mu\left(\nabla_\mu + \nabla_\mu^\star\right) + 2m_0^f - a\nabla_\mu^\star \nabla_\mu\right] \psi^f(x) = a^4 \sum_{x,f} \overline{\psi}^f(x) D_W^f \psi^f(x) \tag{1.15}$$

Here we have introduced the Wilson Dirac operator

$$D_W^f = \frac{1}{2}\left[\gamma_\mu\left(\nabla_\mu + \nabla_\mu^\star\right) + 2m_0^f - a\nabla_\mu^\star\nabla_\mu\right]. \tag{1.16}$$

For convenience we have omitted the color and Dirac indices for the fields as well as the Dirac operator. The flavor f affects the Dirac operator only through the bare quark mass m_0^f.

In order for the Dirac operator to be covariant under gauge transformation we define the covariant derivatives at the point x[1]

$$\nabla_\mu\psi(x) = \frac{1}{a}\left[U_\mu(x)\psi(x+a\hat{\mu}) - \psi(x)\right] \tag{1.17}$$

$$\nabla_\mu^\star\psi(x) = \frac{1}{a}\left[\psi(x) - U_\mu^\dagger(x-a\hat{\mu})\psi(x-a\hat{\mu})\right] \tag{1.18}$$

that include the gauge links $U_\mu(x)$ and $U_\mu^\dagger(x-a\hat{\mu})$.

The first and second terms in the Dirac operator stem directly from the discretization of the theory. The Wilson term, including the double derivative, is defined only on the lattice. It vanishes in the naive continuum limit as $a \to 0$. It is included to remove so-called doublers. Doublers are extra poles in the propagator that correspond to additional, unwanted fermions in the continuum limit. More information can be found in [71, 79]. The chiral symmetry

$$\overline{\psi} \to \overline{\psi}' = \overline{\psi}e^{i\alpha\gamma_5} \tag{1.19}$$

$$\psi \to \psi' = e^{i\alpha\gamma_5}\psi \tag{1.20}$$

$$\overline{\psi}D\psi \to \overline{\psi}'D\psi' = \overline{\psi}\frac{1}{2}\gamma_\mu\left(\nabla_\mu + \nabla_\mu^\star\right)\psi + e^{2i\alpha\gamma_5}\overline{\psi}\left(m_0^f - \frac{a}{2}\nabla_\mu^\star\nabla_\mu\right)\psi \tag{1.21}$$

is broken by the Wilson term, even in the case of zero quark mass $m_0^f = 0$. The symmetry is only restored in the continuum limit.

The Wilson term also introduces $\mathcal{O}(a)$ lattice artifacts, where without the Wilson term the artifacts start at $\mathcal{O}\left(a^2\right)$. These artifacts have been studied [80–82] in order to improve the lattice theory. In [83] Sheikholeslami and Wohlert executed the Symanzik improvement program [80, 81] for the Wilson fermion action and introduced an improvement term.

The improved fermion action S_F is made up of the Wilson action S_W [3] shown in eq. (1.15) and the $\mathcal{O}(a)$ improvement term S_I.

$$S_\mathrm{F} = S_\mathrm{W} + S_\mathrm{I} \tag{1.22}$$

$$S_\mathrm{I} = c_\mathrm{sw}a^5\sum_x\sum_{\mu<\nu}\overline{\psi}(x)\frac{1}{2}\sigma_{\mu\nu}\hat{F}_{\mu\nu}\psi(x) \tag{1.23}$$

The improvement term uses the clover definition of $\hat{F}_{\mu\nu}$ [84]:

$$\sigma_{\mu\nu} = \left[\gamma_\mu, \gamma_\nu\right]/2i \tag{1.24}$$

$$\hat{F}_{\mu\nu}(x) = \frac{-i}{8a^2}\left(Q_{\mu\nu}(x) - Q_{\nu\mu}(x)\right) \tag{1.25}$$

$$Q_{\mu\nu}(x) = U_P^{\mu,\nu}(x) + U_P^{\nu,-\mu}(x) + U_P^{-\mu,-\nu}(x) + U_P^{-\nu,\mu}(x) \tag{1.26}$$

Including the improvement term, the full Dirac operator takes the form

$$D^f = D_W^f + c_\mathrm{sw}\frac{a}{2}\sum_{\mu>\nu}\sigma_{\mu\nu}\hat{F}_{\mu\nu}\delta_{x,y} \tag{1.27}$$

[1] Here the derivative $\nabla\psi$ is evaluated at the point x. A more thorough notation would be $[\nabla\psi](x)$.

The Dirac operator is often expressed in terms of the κ parameter

$$\kappa^f = \frac{1}{2am_0^f + 8}. \tag{1.28}$$

Rescaling the fields ψ_f and $\overline{\psi_f}$ by the constant factor $\sqrt{a\kappa^f}$ we can write the applications of the Dirac operator as

$$\begin{aligned} D^f\psi^f(x) = &\left(1 + c_{\mathrm{sw}}a^2\kappa^f\right)\psi^f(x) - \\ &\kappa^f\sum_\mu \left(\gamma_\mu - 1\right)U_\mu(x)\psi^f(x + a\hat{\mu}) + \left(\gamma_\mu + 1\right)U_\mu^\dagger(x - a\hat{\mu})\psi^f(x - a\hat{\mu}). \end{aligned} \tag{1.29}$$

In typical simulations not all fermion flavors are considered and the sum over the flavor index f in eq. (1.15) is truncated. Details of the simulation employed here are found in chapter 3.

In part II it will be useful to consider the matrix structure of the Dirac operator. The structure is seen more clearly in the following equivalent definition using the Kronecker Delta to show the band structure of the matrix

$$\begin{aligned} D^f(x, y) = &\left(1 + c_{\mathrm{sw}}a^2\kappa^f\right)\cdot\delta_{x,y} - \\ &\kappa^f\sum_\mu \left(\gamma_\mu - 1\right)U_\mu(x)\cdot\delta_{y,x+a\hat{\mu}} + \\ &\kappa^f\sum_\mu \left(\gamma_\mu + 1\right)U_\mu^\dagger(x - a\hat{\mu})\cdot\delta_{y,x-a\hat{\mu}}. \end{aligned} \tag{1.30}$$

1.2.1 Pseudofermion Method

In the fermion path integral formalism we have to evaluate integrals of the form

$$Z_F = \int \mathcal{D}\left[\psi\right]\mathcal{D}\left[\overline{\psi}\right]\exp\left(-S[\psi, \overline{\psi}]\right) = \int \mathcal{D}\left[\psi\right]\mathcal{D}\left[\overline{\psi}\right]\exp\left(-\sum_f \overline{\psi}_f D_f \psi_f\right). \tag{1.31}$$

The fields ψ and $\overline{\psi}$ are Grassmann-valued. Non commuting Grassmann variables are difficult to deal with in numerical simulations. We can, however, integrate over the Grassmann variables to get

$$Z_F = \prod_f \det\left(D_f\right). \tag{1.32}$$

If we simulate two quark flavors with degenerate mass we can simplify the expression

$$Z_F = \det\left(D_u\right)\det\left(D_d\right) = \det\left(\gamma_5 D\gamma_5\right)\det\left(D\right) = \det\left(D^\dagger D\right), \tag{1.33}$$

since $\det\left(\gamma_5\right) = 1$. Note that $D^\dagger D$ is positive definite. In the pseudo fermion method devised in [85] the fermion determinant is evaluated using the integral over complex vectors ϕ.

$$\det\left(D^\dagger D\right) = \frac{1}{Z_\phi}\int \mathcal{D}\left[\phi^\dagger\right]\mathcal{D}\left[\phi\right]\exp\left(-\phi^\dagger\left(D^\dagger D\right)^{-1}\phi\right) \tag{1.34}$$

$$Z_\phi = \int \mathcal{D}\left[\phi^\dagger\right]\mathcal{D}\left[\phi\right]\exp\left(-\phi^\dagger\phi\right) \tag{1.35}$$

The vector ϕ is made up of four complex spinor components that consist of complex three vectors. In total each spinor contains 24 degrees of freedom.

The pseudo fermion action

$$S_{\mathrm{PF}} = \phi^\dagger \left(D^\dagger D\right)^{-1} \phi \tag{1.36}$$

can now be added to the gauge action and sampled using Monte Carlo techniques [78]. We have avoided the necessity to numerically implement Grassmann variables. We do, however, need to solve the hermitian Dirac system $\phi = D^\dagger D \chi$ for every evaluation of the pseudo fermion action.

In the following we will refer to spinors with the letter ψ, as it is common. Note that instead of a vector of Grassmann variables ψ is a complex pseudo fermion vector.

Part I

Scale Setting

2 | Introduction

Lattice QCD simulations as described in chapter 1 are only able to provide estimates of dimensionless quantities. To compare results between different simulations and obtain results in physical units the scale of the lattice simulation needs to be set using experimental input. A comprehensive overview of scale setting in lattice QCD simulations can be found in [55]. Several key points are repeated here.

Combinations of lattice observables with a well-defined continuum limit are the predictions of the theory. We can for example chose to compare all masses to the mass of the proton and calculate ratios

$$R_i = \frac{m_i}{m_{\text{proton}}} \tag{2.1}$$

on the lattice. In contrast to the mass m_i itself, the ratio has a well-behaved continuum limit

$$R_i^{\text{cont}} = \lim_{a \to 0} R_i \tag{2.2}$$

We can then use an experimental measurement of the proton mass $m_{\text{proton}}^{\text{exp}}$ to extract the dimensionful, physical mass m_i from the lattice measurement

$$m_i^{\text{phys}} = R_i^{\text{cont}} m_{\text{proton}}^{\text{exp}}. \tag{2.3}$$

In this example the proton mass is used as a reference to set the scale of the lattice simulation.

In the context of lattice simulations this is often considered equivalent to calculating the lattice spacing a for each considered coupling g_0. The lattice spacing is determined using an experimental measurement $m_{\text{proton}}^{\text{exp}}$ of the scale as well as the lattice measurement of am_{proton}

$$a = \frac{(am_{\text{proton}})^{\text{lat}}}{m_{\text{proton}}^{\text{exp}}}. \tag{2.4}$$

Once the lattice spacing is fixed, other observables can be calculated in physical units using

$$m_i^{\text{lat}} = \frac{(am_i)^{\text{lat}}}{a}. \tag{2.5}$$

We can now take the continuum limit of m_i^{lat} to get the physical value.

The reference scale itself loses its predictive power. This is most clearly seen in eqs. (2.1) and (2.3). The ratio $R_{\text{proton}} \equiv 1$ and eq. (2.3) only yields the experimental result. Once the scale is set, however, we can extract predictions of other observables from lattice simulations. Since the reference scale is used for all dimensionful lattice observables, it is very important for lattice QCD simulations. Additionally, when planning the simulations the quark masses in the Lagrangian (eq. (1.1)) have to be set. They are tuned to a line of constant physics along which the continuum limit can be taken. The line of constant physics is defined using a number of dimensionless observables which are fixed to constant values. This ensures that the underlying physics are not changed as the continuum limit is taken. To define the line of constant physics one uses a combination of observables that depend strongly on the quark masses and the lattice scale.

A variety of observables can be used for scale setting. The following requirements, however, narrow the selection down to a few popular choices. The scale should be easy to calculate on the lattice allowing its determination with high statistical precision. On the lattice we need to be able to understand and control the systematic uncertainties. These uncertainties can originate from finite size effects, contamination by excited states and chiral as well as continuum extrapolations. It is evidently beneficial to choose a scale setting quantity where these effects are small and well studied. Additionally, the dependence on the quark masses should be weak. As we will see in section 5.2, there will be mistunings from the chiral trajectory. If the dependence of the scale setting quantity on the quark masses is weak, the effect of these mistunings is small. The tuning can then be done somewhat independent of the scale setting.

In the following we discuss several choices of scale setting observables as well as their advantages and disadvantages. Many of the arguments are taken from [55]. We will start with phenomenological scales whose physical value can be determined experimentally before moving to intermediate scales, where the physical value is not known from the experiment. The physical value of intermediate scales, sometimes also called theory scales, is determined in conjunction with one of the phenomenological scales.

Phenomenological Scales

Phenomenological scales are observables that can be measured in the lattice as well as experimentally. They are physical observables. Several common choices are given below.

Baryon Masses m_p, m_Ω
The masses of baryons such as the Proton or the Omega baryon can, in principle, also be used for scale setting. The proton mass is favorable from an experimental perspective as the measured precision is extremely high [86]. However, the lattice determination of baryon masses suffers from a signal-to-noise problem that makes it hard to determine the baryon masses accurately [64, 87]. The statistical uncertainties are smaller for the Omega baryon, which has been used to set the scale of lattice simulations [56, 88–90]. The Omega baryon has a weak dependence on the light quark mass and a strong dependence on the strange quark mass. This makes it suitable for trajectories where the strange quark mass is kept constant. On the trajectory defined by a constant sum of the quark masses the Omega mass is not well suited for scale setting purposes.

Meson Masses m_ρ, m_Υ
The masses for the Pion and Kaon are often used to define the line of constant physics. In this case they are no longer available to set the scale. In quenched simulations of the past, the ρ mass has been used to set the scale [91–93]. Using dynamical fermions the ρ meson becomes unstable and is therefore not used for scale setting. There are groups [94, 95] that use the mass of the Υ meson to set the scale because of its precise experimental determination. The dependence on the b quark, however, potentially introduce large discretization effects. For this reason it is not a popular choice for scale setting observables and is not used here.

Meson Decay Constants f_π, f_K
The mesonic decay constants are a popular choice for scale setting [1, 57, 96]. In particular,

they are used to calculate the physical values of the intermediate scales $r_0, t_0, w_0, \ldots$ in physical units. On the lattice the decay constants can be extracted from correlator measurements. The measurements exhibit a large plateau which indicates that excited state contributions decay fast and the values can be calculated precisely. The physical measurements, however, are determined using a weak process $\pi/K \to l\nu$. This leads to a measurement of the product $V_{ud}f_\pi$ and $V_{us}f_K$, which depends on the CKM matrix elements V_{ud}, V_{us}. The determination of the CKM matrix elements introduces a new source of uncertainties increasing the error by 1% for the Pion (V_{ud}) and by 73% for the Kaon (V_{us}) decay constant measurements [9].

Intermediate Scales

The line of constant physics can be defined using intermediate scales whose value does not need to be known experimentally. One can, for example, define the line of constant physics by keeping the dimensionless quantities $\sqrt{t_0}m_\pi$ and $\sqrt{t_0}m_K$ constant. Here the flow scale t_0 acts as an intermediate scale. Note that it is not necessary to know the physical value of the scale t_0 to tune the ensembles to a line of constant physics. Only if we want this line to pass though the physical point we have to determine the physical value of the scale t_0. To do that we have to use one of the phenomenological scales described above. We can then calculate the dimensionless combination $\sqrt{t_0}S_\text{phen}$ of the intermediate scale t_0 and the phenomenological scale S_phen on the lattice and evaluate the ratio with the experimental value of the phenomenological scale.

$$\sqrt{t_0^\text{phys}} = \frac{\left(\sqrt{t_0}S_\text{phen}\right)^\text{lat}}{S_\text{phen}^\text{exp}} \tag{2.6}$$

We can now ensure that the line of constant physics defined by t_0^phys runs through the physical point.

In the following we list several intermediate scales.

Force scale r_0

The scale r_0 [97–99] is derived from the static quark antiquark potential. It is the distance r where the force between two quarks takes a certain value

$$r^2 F(r)\big|_{r=r_c} = c, \quad r_0 \equiv r|_{c=1.65}\,. \tag{2.7}$$

The physical value of the scale is not known experimentally, but as discussed above this is not necessary, if the scale is used in conjunction with a phenomenological scale. The force $F(r)$ is derived from the static quark potential which is calculated using the evaluation of Wilson loops. The static quark potential is studied in [97, 100]. The potential forms a plateau very early [55] indicating that excited state effects are small.

Flow Scale t_0, w_0

The gradient flow scale t_0 [101, 102] is discussed in detail in sections 3.3.1 and 4.1. The gradient flow is a smoothing operation on the gauge fields with a mean radius of t. The scale t_0 is defined implicitly using the energy density $E(t)$ and the Wilson flow time t

$$t^2 E(t)\big|_{t=t_c} = c, \quad t_0 \equiv t|_{c=0.3}\,. \tag{2.8}$$

It shares many of the properties of the force scale r_0 but is more easily calculated to high precision. As such, it is a popular choice for scale setting [1, 57, 58, 103]. The systematic uncertainties are studied in detail [104–106]. The gradient flow scale can be used as an intermediate scale comparing different lattice ensembles. To fully set the scale of the lattice simulation, the flow scale has to be supplemented with a scale whose continuum limit is defined such that it can be compared to physical measurements. This strategy is elaborated in [1] and chapter 5.

Some groups [56, 57, 103] prefer using the scale w_0 that is defined using the slope of $t^2E(t)$.

3 | Simulation

In this section we will briefly describe the lattice QCD simulations. A detailed introduction and analysis of lattice QCD simulations can be found in [71, 78, 79, 107]. The simulation is carried out by the generation of gauge ensembles [78] and the subsequent measurement of observables on these ensembles. Here, we will take the gauge ensembles for granted and focus instead on the measurement and analysis of lattice observables.

We will start by presenting the parameters of the simulation. We will then give an overview of the measured observables relevant for this analysis in section 3.3. In sections 3.2 and 3.5 we will discuss how to correct several algorithmic and finite size artifacts. Finally, we will give a short introduction on the statistical error analysis in sections 3.5 and 3.6.

3.1 Ensembles

In chapter 1 we briefly introduced gauge configurations that are used to evaluate the expectation values of lattice operators. Configurations that stem from the same Monte Carlo chain with fixed parameters are called a gauge ensemble. The generation of these gauge ensembles is very computationally intensive. For that reason ensembles are often generated once and stored. In this way many different analyses can be performed on the same gauge configurations.

In this work, we are using the ensembles generated by the CLS group. An overview of the ensembles and their generation can be found in [78, 108]. The CLS ensembles used here simulate two degenerate light quarks and a strange quark using non-perturbatively improved Wilson fermions and a tree level improved gauge action. The ensembles differ in their size, the lattice spacing mediated by the inverse coupling $\beta = \frac{6}{g_0^2}$ and the simulated bare quark masses that are set using the κ parameter (see eq. (1.28)). In table 3.1 we give an overview of the ensembles used in this analysis. We use ensembles with five different lattice spacings represented by the different values of the inverse coupling β. For each lattice spacing the trajectory has been attached to the symmetric point where $\kappa_l = \kappa_s$. The trajectory with constant sum of the bare quark masses $\mathrm{tr}\,(M) = 2m_l + m_s = \text{const.}$ extends left towards the point of physical quark masses. This chiral trajectory will be discussed in section 5.2. The ensemble landscape is shown in fig. 3.1. Each lattice spacing is represented by a different color. The symmetric ensembles are to the right of the plot. Using these ensembles we can take the continuum limit towards $a \to 0$ as well as the limit toward physical bare quark masses. Ensembles at different lattice spacings are tuned in such a way that the quantities $t_0 m_\pi^2$ and $t_0(m_K^2 + m_\pi^2/2)$ are constant defining the line of constant physics (LOC) along which the continuum limit is taken as well as the chiral trajectory that dictates the extrapolation to physical quark masses. Different ensembles with the same lattice spacing are tuned such that the improved bare coupling $\tilde{g}_0^2 \sim \frac{1}{\beta}$ is constant along the chiral trajectory towards the physical point.

The size of the lattices is chosen such that $m_\pi L \gtrsim 4$ and finite volume effects are small. The

name	id	runs	L	T	β	κ_l	κ_s	$m_\pi L$	MDU
H101	0	0,1	32	96	3.40	0.136760	0.136760	5.9	8064
H102	0	1	32	96	3.40	0.136865	0.136549	4.9	4116
H102	1	2	32	96	3.40	0.136865	0.136549	5.0	4032
H105	0	1,2	32	96	3.40	0.136970	0.136341	3.9	8276
N101	2	3,4,5,6	48	128	3.40	0.136970	0.136341	5.9	5266
C101	1	14	48	96	3.40	0.137030	0.136222	4.6	8000
B450	0	0	32	64	3.46	0.136890	0.136890	5.2	6448
S400	0	0,1	32	128	3.46	0.136984	0.136702	4.3	11492
D450	1	10	64	128	3.46	0.137126	0.136420	5.4	2000
D452	0	2	64	128	3.46	0.137164	0.136346	3.8	4000
N202	0	1,2	48	128	3.55	0.137000	0.137000	6.4	7608
N203	0	0,1	48	128	3.55	0.137080	0.136840	5.4	6172
N200	0	0,1	48	128	3.55	0.137140	0.136721	4.4	6848
D200	0	0	64	128	3.55	0.137200	0.136602	4.1	8004
E250	1	1	96	192	3.55	0.137233	0.136537	4.0	4036
N300	0	1	48	128	3.70	0.137000	0.137000	5.1	2028
N300	1	2	48	128	3.70	0.137000	0.137000	5.1	6162
N302	0	1	48	128	3.70	0.137064	0.136872	4.2	8804
J303	0	3	64	192	3.70	0.137123	0.136755	4.2	8584
E300	1	1	96	192	3.70	0.137163	0.136675	4.2	4556
J500	0	4,5	64	192	3.85	0.136852	0.136852	5.2	6312
J501	0	1,2	64	192	3.85	0.136903	0.136750	4.2	6540

Table 3.1: Information about CLS [78, 108] gauge ensembles. We give the id and the runs that make up one ensemble followed by the parameters of the specific ensemble: spacial and temporal extent in lattice units, the inverse coupling $\beta = \frac{6}{g_0^2}$, the κ parameters for the light and strange quarks. The column labeled $m_\pi L$ lists the Pion mass times the spacial extent of the lattice as a reference to the physical size of the lattice. The last column lists the length of the Monte Carlo chain in molecular dynamics units.

lattice extent in this unit is given in the second to last column of table 3.1. At the boundary in the spacial directions the lattice is continued periodically. The euclidean time axis uses either periodic or open boundary conditions. A “5” in the third place of the ensembles name indicates periodic boundary conditions in time. Open boundary conditions are used because they are able to move more freely between individual sectors of the configuration space, thus improving the sampling of the path integral [73]. The size of the ensemble is given in molecular dynamics units (MDU) in the last column of table 3.1. A configuration is exported every four MDUs. The total number of configurations of the ensemble is therefore a quarter of the number of MDUs. The improvement coefficient c_{sw} used in eq. (1.23) is calculated non-perturbatively in [109].

Within the CLS group there are replica runs consisting of multiple Monte Carlo Chains with identical parameters. The size of such replica runs in MDU has been added. For this analysis we have grouped such runs under a common ID as indicated by the second and third column of table 3.1.

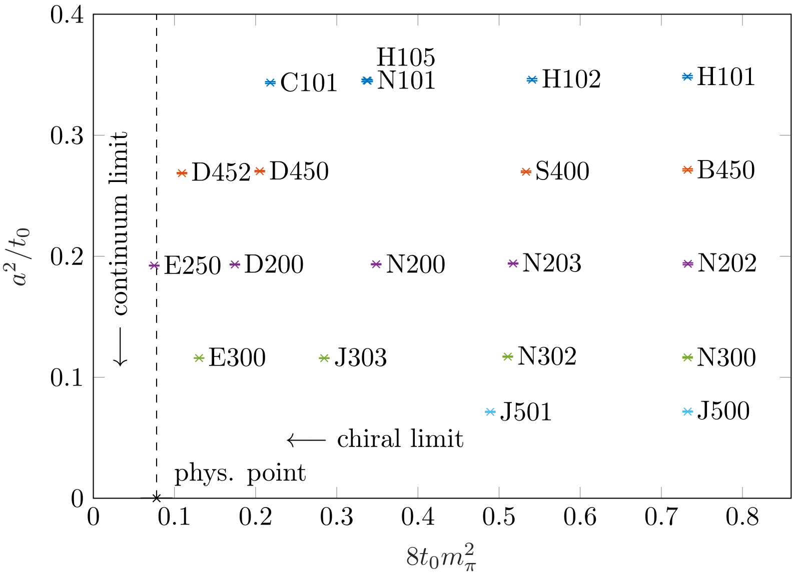


Figure 3.1: Parameter landscape of the CLS [78, 108] ensembles. The rightmost ensembles in each row represent the symmetric points where $m_\pi = m_K$. Each row represents the trajectory with constant sum of the quark masses $\mathrm{tr}\,(M) = 2m_l + m_s = \text{const}$. The inverse coupling indicated by different colors runs from $\beta = 3.4$ for the coarsest lattices on the top to $\beta = 3.85$ for the finest lattices on the bottom. The chiral limit towards physical point runs along x axis, the continuum limit $a \to 0$ along the y axis.

3.2 Reweighting

CLS chose to run the generation of the ensembles at different actions than the ones discussed in chapter 1. Expectation values $\langle X\rangle$ in the target theory can then be calculated from expectation values $\langle\cdots\rangle_W$ in the modified theory using reweighting.

$$\langle X\rangle = \frac{\langle X\cdot W\rangle_W}{\langle W\rangle_W}. \tag{3.1}$$

We include three types of reweighting factors that have been calculated for previous analyses [1, 110]. The final reweighting factor is the product of the individual factors

$$W = W_0 W_1 W_2. \tag{3.2}$$

Twisted Mass Reweighting

The twisted mass reweighting factor stems from shifting the spectrum of the Dirac operator. The method was devised in [111] and is summarized in [78]. It is done to overcome the $\det D = 0$ barriers in the configuration space that could otherwise not be crossed by the Hybrid Monte Carlo algorithm. The reweighting factor is given by

$$W_0 = \det\left(\frac{\left(\hat{Q}^2+\mu_0^2\right)\hat{Q}^2}{\left(\hat{Q}^2+\mu_0^2\right)^2}\right) \tag{3.3}$$

where $\hat{Q}$ is the Schur complement

$$\hat{Q} = Q_{\mathrm{ee}} - Q_{\mathrm{eo}}Q_{\mathrm{oo}}^{-1}Q_{\mathrm{oe}} \tag{3.4}$$

of the even-odd preconditioned [112] hermitian Dirac operator $Q = \gamma_5 D$.

RHMC Reweighting

During the generation of the ensembles the strange quark is simulated using RHMC algorithm [113, 114]. In the RHMC algorithm the fermion determinant is approximated using a rational function. The reweighting factor W_1 is used to remedy this approximation. Its definition is given in [78].

Sign Reweighting

The generation of the ensembles assumed that the sign of the strange quark determinant is always positive. Recent findings conclude that this is not always the case [110]. Ensembles with a high number ($> 10\%$) of configurations with negative strange quark determinant have been removed from this analysis. For the other ensembles we have included a reweighting factor

$$W_2 = \frac{\det D_{\mathrm{s}}}{|\det D_{\mathrm{s}}|} \tag{3.5}$$

that corrects this sign.

In the following the expectation value $\langle X\rangle$ will refer to the fully reweighted observables.

3.3 Observables

On the CLS ensembles discussed in section 3.1 we can now calculate the observables used to set the scale. Primary observables that are measured directly on the gauge configurations are the flow scale t_0 and mesonic correlation functions. These observables are subject to reweighting presented in section 3.2. From the correlation functions we can extract the pseudoscalar masses and decay constants as well as the PCAC mass. The pseudoscalar observables are corrected for finite volume effects as shown in section 3.5. In the following sections we will define these observables and their measurement on the lattice.

3.3.1 Gradient Flow Scale

The gradient flow is a form of controlled smearing of the gauge fields [101, 115]. The original gauge field $U(x, \mu)$ is evolved along the newly introduced flow time t. The evolution is defined by the following flow equation

$$\partial_t V_t(x,\mu) = -g_0^2 \left\{\partial_{x,\mu} S_W(V_t)\right\} V_t(x,\mu), \quad V_t(x,\mu)|_{t=0} = U(x,\mu) \tag{3.6}$$

with the Wilson gauge action defined in eq. (1.10).

Numerically integrating the flow equation (3.6) yields the t-dependent gauge fields $V_t(x, \mu)$. The Wilson flow can be considered a smoothing operation with the mean-square radius $\sqrt{8t}$ [101]. Consequently, the flow time has dimension $[t] = \text{length}^2 \sim \frac{1}{\text{MeV}^2}$. It will be used later to construct dimensionless observables (see section 5.1).

Observables such as the energy density can be expressed as functions of the transformed field $V_t(x, \mu)$. Doing this results in a quantity that is finite and does not need to be renormalized for positive flow times $t > 0$. The energy density can be defined using the clover definition of $F_{\mu\nu}$ given in eqs. (1.25) and (1.26).

$$E(x,t) = a^4 \sum_{\mu,\nu} \mathrm{tr}\left(\hat{F}_{\mu\nu}(x)\hat{F}_{\mu\nu}(x)\right) \tag{3.7}$$

The links in the clover definition of $\hat{F}_{\mu\nu}(x)$ are replaced with the smeared links $V_t(x, \mu)$ from the gradient flow. The average energy density

$$E(t) = \frac{1}{V}\sum_x E(x,t) \tag{3.8}$$

depends only on the flow time t.[1]

The quantity $\sqrt{t^2 E(t)}$ is a constant proportional to the renormalized gauge coupling in leading order perturbation theory. Since it can be easily and precisely calculated on the lattice, it is a good candidate to set the scale. To this end we define the scale t_0 implicitly by

$$\left\langle t^2 E(t)\right\rangle\big|_{t=t_0} = 0.3 \tag{3.9}$$

as proposed in [101]. Details on the measurement of the flow scale can be found in [1, 78].

[1] The sum over the entire volume is only correct for periodic boundary conditions. For open boundary conditions the average is taken over a plateau region away from the boundary. See section 4.1 for details

3.3.2 Correlators

The analysis is centered around non-singlet two-point correlation functions between pseudoscalar

$$P^{rs}(x) = \overline{\psi}^r(x)\gamma_5\psi^s(x) \tag{3.10}$$

and axial

$$A_\mu^{rs}(x) = \overline{\psi}^r(x)\gamma_\mu\gamma_5\psi^s(x) \tag{3.11}$$

fermion currents. The correlators relevant here are defined by

$$\begin{aligned} f_P^{rs}(x_0, y_0) &= -\frac{a^6}{L^3}\sum_{\vec{x},\vec{y}} \langle P^{rs}(x_0,\vec{x})P^{sr}(y_0,\vec{y})\rangle \\ f_A^{rs}(x_0, y_0) &= -\frac{a^6}{L^3}\sum_{\vec{x},\vec{y}} \langle A_0^{rs}(x_0,\vec{x})P^{sr}(y_0,\vec{y})\rangle\,. \end{aligned} \tag{3.12}$$

After executing the Wick contractions, the two-point functions are expressed as

$$f_X^{rs}(x_0, y_0) = -\frac{a^6}{L^3}\sum_{\vec{x},\vec{y}} \langle \mathrm{tr}\left(\Gamma_X S^r(x,y)\gamma_5 S^s(y,x)\right)\rangle \tag{3.13}$$

with the propagators $S^f(x,y)$ and

$$\Gamma_X = \begin{cases} \gamma_5 & , X = P \\ \gamma_0\gamma_5 & , X = A_0. \end{cases} \tag{3.14}$$

The axial correlator is $\mathcal{O}(a)$ improved by

$$f_A^{rs}(x_0, y_0) \to f_A^{rs}(x_0, y_0) + ac_A\tilde{\partial}_{x_0} f_P^{rs}(x_0, y_0) \tag{3.15}$$

according to the Symanzik improvement program [80, 81]. The improvement term uses the symmetric derivative in the time direction

$$\tilde{\partial}_0 = \frac{1}{2}\left(\partial_0 + \partial_0^\star\right). \tag{3.16}$$

The coefficient c_A in eq. (3.15) is determined non-perturbatively in [116].

The measurements are taken at a fixed source time $y_0 = a, 2a, T-a, T-2a$ and evaluated at all sink times x_0. In the following we will often omit the source position from the argument of the correlators and refer to them as $f_P(x_0)$. Details on the measurements of the correlators are presented in section 4.2.

3.3.3 Pseudoscalar Mass

The pseudoscalar mass can be extracted from the pseudoscalar correlation function defined in section 3.3.2. A detailed analysis of the correlation function is given in [117]. Several key steps are repeated here.

With the help of the transfer matrix the correlator $f_X(x_0, y_0)$ can be expressed in terms of the eigenstates of the Hamilton operator

$$H\left|\alpha, n\right\rangle = E_n^\alpha \left|\alpha, n\right\rangle. \tag{3.17}$$

Here n indicates the energy level and α labels all other quantum numbers. Inserting a full set of Hamiltonian eigenstates and assuming open boundary conditions, the correlation function reads

$$f_X(x_0, y_0) = \sum_{\alpha,\beta} \sum_{n,m} \frac{\langle \Omega \,|\, \beta, m\rangle}{\langle \Omega \,|\, 0, 0\rangle} e^{-E_m^\beta (T-x_0)} \langle \beta, m \,|\, X \,|\, \alpha, n\rangle \, e^{-E_n^\alpha (x_0-y_0)} \langle \alpha, n \,|\, \phi_\pi(y_0)\rangle \,. \tag{3.18}$$

We can substitute $X = P$ for the pseudoscalar correlator and $X = A_\mu$ for the axial correlator. The boundary state is indicated by $|\Omega\rangle$. The state at the source location y_0 is indicated by

$$|\phi_\pi(y_0)\rangle = L^3 \sum_{\gamma,l} P \,|\gamma, l\rangle \, e^{-y_0 E_l^\gamma} \frac{\langle \gamma, l \,|\, \Omega\rangle}{\langle 0, 0 \,|\, \Omega\rangle}. \tag{3.19}$$

For large distances $|x_0 - y_0| \to \infty$ the only state propagating between X and $\phi_\pi(y_0)$ is the Pion state $|\pi, 0\rangle$. Because of the composition of the boundary states only the terms with $\beta = 0$, i.e. states with zero momentum, survive: $\sum_{\beta,m} \langle \Omega \,|\, \beta, m\rangle = \sum_m \langle \Omega \,|\, 0, m\rangle$. With these simplifications the correlation function can be expressed as

$$f_X(x_0, y_0) = \left[1 + \eta_X^T e^{-E_1^0 (T-x_0)} + \cdots\right] \langle 0, 0 \,|\, X \,|\, \pi, 0\rangle \, e^{-m_\pi (x_0-y_0)} \langle \pi, 0 \,|\, \phi_\pi(y_0)\rangle \tag{3.20}$$

with

$$\eta_X^T = \frac{\langle \Omega \,|\, 0, 1\rangle \langle 0, 1 \,|\, X \,|\, \pi, 0\rangle}{\langle \Omega \,|\, 0, 0\rangle \langle 0, 0 \,|\, X \,|\, \pi, 0\rangle}. \tag{3.21}$$

The pseudoscalar mass can be extracted from the pseudoscalar correlation function in two ways. For open boundary conditions the pseudoscalar correlation function takes the form

$$f_P^{\text{obc}}(x_0, y_0) = A_1(y_0) e^{-m_{\text{PS}} x_0} + A_2(y_0) e^{-m' x_0} + B_1(y_0) e^{-(E_{2\text{PS}} - m_{\text{PS}})(T-x_0)} + \cdots . \tag{3.22}$$

For periodic ensembles it assumes the symmetric form

$$f_P^{\text{pbc}}(x_0, y_0) = \tilde{A}_1(y_0) \left(e^{-m_{\text{PS}} x_0} + e^{-m_{\text{PS}}(T-x_0)}\right) + \tilde{A}_2(y_0) \left(e^{-\tilde{m}' x_0} + e^{-\tilde{m}'(T-x_0)}\right) + \cdots . \tag{3.23}$$

The pseudoscalar mass appears in the exponent of the first term. We can therefore extract it from a fit to the correlation function. This is done for the periodic ensembles. Details can be found in section 4.3.1.

In some cases it can be difficult to find the optimal fit for the functions above. To avoid complicated fitting procedures, for ensembles with open boundary conditions, we can define the effective mass from two neighboring time slices x_0 and $x_0 + a$.

$$am_{\text{eff}}(x_0) = \log\left(\frac{f_P(x_0)}{f_P(x_0 + a)}\right) = am_{\text{PS}} \left(1 + c_1 e^{-E_1 x_0} + c_2 e^{-E_{2\text{PS}}(T-x_0)} + \cdots\right) \tag{3.24}$$

The exponents of the higher contributions m', E_1 and $E_{2\text{PS}}$ are discussed in section 4.3.

We now have a value for the effective mass for each time slice x_0 and can average these values over a plateau region in the center. How to find a suitable plateau region where the excited state and boundary effects are small is discussed in section 4.3. This approach is more numerically stable than the direct fit method if the slope of the correlation function is large. For periodic correlation functions the slope close to $x_0 \approx T/2$ is very small. In this region uncertainties in the correlation function get amplified. For this reason we choose the effective mass method only for ensembles with open boundary conditions.

3.3.4 PCAC Mass

The bare PCAC mass is defined by the pseudoscalar and axial currents

$$m^{rs}_{\text{PCAC}}(x_0) = \frac{\tilde{\partial}_0 f^{rs}_A(x_0)}{2 f^{rs}_P(x_0)} \tag{3.25}$$

using the improved axial current from eq. (3.15) and the symmetric derivative in time direction from eq. (3.16). The bare PCAC mass is calculated in the same way for both open and periodic boundary conditions.

3.3.5 Pseudoscalar Decay Constant

The pseudoscalar decay constants are extracted from the pseudoscalar and axial correlation functions. An expansion of the correlation functions around the lowest state is given in eq. (3.20).

To be able to extract the matrix element $\langle 0,0 \,|\, X \,|\, \pi, 0\rangle$ we need to eliminate the state $|\phi_\pi(y_0)\rangle$. We can do so using the pseudoscalar correlator

$$L^3 f_P(T-y_0, y_0) = e^{-m_\pi(T-2y_0)} \langle \phi_\pi(y_0) \,|\, \pi, 0\rangle \langle \pi, 0 \,|\, \phi_\pi(y_0)\rangle \,. \tag{3.26}$$

We normalize the correlator by

$$\langle 0,0 \,|\, X \,|\, \pi, 0\rangle = \left[\frac{|f_X(x_0,y_0) f_X(x_0,T-y_0)|}{L^3 f_P(T-y_0,y_0)}\right]^{1/2} + \mathcal{O}\left(e^{-E^0_1(T-x_0)}\right) + \mathcal{O}\left(e^{-(E^\pi_1 - m_\pi)x_0}\right). \tag{3.27}$$

For ensembles with open boundary conditions we calculate the ratio R_{PS} on every time slice x_0.

$$R^{\text{obc}}_{\text{PS}}(x_0) = \left[\frac{f_A(x_0,y_0) f_A(x_0,T-y_0)}{f_P(T-y_0,y_0)}\right]^{1/2} \tag{3.28}$$

We must use the same ratio for ensembles with periodic boundary conditions. This is necessary to get the same behavior as we approach the continuum. Later in the procedure ensembles with open and periodic boundary conditions are analyzed side by side. The periodic correlators are fitted with

$$F_P = C_P\left(e^{-m_{\text{PS}}x_0} + e^{-m_{\text{PS}}(T-x_0)}\right) \tag{3.29}$$

$$F_A = C_A\left(-e^{-m_{\text{PS}}x_0} + e^{-m_{\text{PS}}(T-x_0)}\right). \tag{3.30}$$

The pseudoscalar mass is taken from a previous determination and only the linear coefficients C_P and C_A are determined by the fit. From these we calculate the ratio

$$R^{\text{pbc}}_{\text{PS}} = \frac{C_A}{\sqrt{C_P}}. \tag{3.31}$$

For periodic ensembles we have to specify the fit interval for the correlators. For ensembles with open boundary conditions we determine a plateau region over which the ratio $R^{\text{obc}}_{\text{PS}}(x_0)$ is averaged. These ranges are discussed in section 4.3.

Using the average pseudoscalar ratio $\overline{R_{\text{PS}}}$ we calculate the bare pseudoscalar decay constant according to

$$f^{\text{bare}}_{\text{PS}} = \sqrt{\frac{2}{m_{\text{PS}}}}\,\overline{R_{\text{PS}}}. \tag{3.32}$$

β	Z_A
3.40	0.75642(72)
3.46	0.76169(93)
3.55	0.76979(43)
3.70	0.78378(47)
3.85	0.79667(47)

Table 3.2: Renormalization factor Z_A for different values of β. The values are calculated in [118, tab.7] using the chirally rotated Schrödinger functional, the "l-convention" and subtraction of the one-loop cutoff effects.

We use the mass measured earlier on the same ensemble.

We renormalize the bare decay constant using the renormalization factor Z_A non-perturbatively determined in [118] and listed in table 3.2 as well as the perturbative improvement coefficients $\bar{b}_A$ and $\tilde{b}_A$ from [119].

$$f_{\mathrm{PS}} = Z_A(\tilde{g}_0)\left[1 + \bar{b}_A a\mathrm{tr}\left(M_q\right) + \tilde{b}_A a m^{rs}_{\mathrm{PCAC}}\right] f^{\mathrm{bare}}_{\mathrm{PS}} \tag{3.33}$$

$$\tilde{b}_A = 1 + 0.0472 g_0^2 + \mathcal{O}\left(g_0^4\right) \tag{3.34}$$

$$\bar{b}_A = \mathcal{O}\left(g_0^4\right) \tag{3.35}$$

3.4 Mass Derivatives of Observables

We also measure the derivatives of the primary observables with respect to the bare quark masses [1, 59–63]. The derivatives are needed to correct the mistuning of the ensembles. This procedure is introduced in [1] and discussed in section 5.2.

The quantities introduced in sections 3.3.1 to 3.3.5 are functions of expectation values of primary observables. As a first step, let us consider the mass derivatives of expectation values of a primary observable

$$\overline{A} = \langle A \rangle = \frac{1}{Z}\int A e^{-S} dU \tag{3.36}$$

$$Z = \int e^{-S} dU. \tag{3.37}$$

The derivative of the expectation value can be split in three components

$$\begin{aligned}\frac{d\overline{A}}{dm_0^f} &= \frac{1}{Z}\int \frac{\partial A}{\partial m_0^f} e^{-S} dU + \frac{1}{Z}\int A \frac{\partial\left(e^{-S}\right)}{\partial m_0^f} dU - \frac{1}{Z^2}\int A e^{-S} dU \int \frac{\partial\left(e^{-S}\right)}{\partial m_0^f} dU \\ &= \left\langle \frac{\partial A}{\partial m_0^f}\right\rangle - \left\langle \frac{\partial S}{\partial m_0^f} A\right\rangle + \left\langle \frac{\partial S}{\partial m_0^f}\right\rangle \langle A \rangle .\end{aligned} \tag{3.38}$$

The first term is the partial derivative of the observable itself and the second and third part stem from the derivative of the action used in the calculation of the expectation value.

For a function of expectation values $f(\overline{A}_i)$ the derivative is given by

$$\begin{aligned}\frac{df(\overline{A}_i)}{dm_0^f} &= \sum_i \frac{\partial f}{\partial \overline{A}_i}\frac{d\overline{A}_i}{dm_0^f} \\ &= \sum_i \frac{\partial f}{\partial \overline{A}_i}\left[\left\langle \frac{\partial A_i}{\partial m_0^f}\right\rangle - \left\langle \frac{\partial S}{\partial m_0^f}A_i\right\rangle + \left\langle \frac{\partial S}{\partial m_0^f}\right\rangle \langle A_i\rangle\right].\end{aligned} \tag{3.39}$$

The Wilson fermion action is given in eq. (1.15). Derivation with respect to the bare quark mass with flavor f results in

$$\frac{\partial S}{\partial m_0^f} = a^4 \sum_x \overline{\psi}_f(x)\psi_f(x). \tag{3.40}$$

The explicit mass derivatives of the observables $\frac{\partial A}{\partial m_0^f}$ need to be calculated for each individual observable A using the definitions in section 3.3.

3.5 Finite Volume Effects

Previously we analyzed boundary terms in the direction of the time x_0 and assumed an infinite spacial extent of the lattice. In reality the spacial coordinates of the lattice are repeated periodically at a distance L. The effects of this cutoff are studied in [120, 121]. The authors argue that the finite size effects for the pseudoscalar masses and decay constants are below 1%, if the box is bigger than $L > 2\,\mathrm{fm}$ and $m_\pi L > 1$. Here, the smallest lattices are $L = 2.35\,\mathrm{fm}$ and $m_\pi L = 3.9$. Accordingly, the average correction for the observables used here is 0.1% with the biggest correction at 0.6%.

To apply the corrections, the authors of [120, 121] calculate the ratios

$$R_X = \frac{X(L) - X}{X} \tag{3.41}$$

between the finite volume ($X(L)$) and infinite volume (X) quantities for various observables X. Here, we use these ratios to correct the finite volume effects in the measurements of the Pion and Kaon masses and decay constants and their quark mass derivatives. In chiral perturbation theory the ratios are calculated [120, 121] as follows

$$R_{m_\pi} = \frac{1}{4}\xi_\pi \tilde{g}_1(\lambda_\pi) - \frac{1}{12}\xi_\eta \tilde{g}_1(\lambda_\eta) \tag{3.42}$$

$$R_{m_K} = \frac{1}{6}\xi_\eta \tilde{g}_1(\lambda_\eta) \tag{3.43}$$

$$R_{f_\pi} = -\xi_\pi \tilde{g}_1(\lambda_\pi) - \frac{1}{2}\xi_K \tilde{g}_1(\lambda_K) \tag{3.44}$$

$$R_{f_K} = -\frac{3}{8}\xi_\pi \tilde{g}_1(\lambda_\pi) - \frac{3}{4}\xi_K \tilde{g}_1(\lambda_K) - \frac{3}{8}\xi_\eta \tilde{g}_1(\lambda_\eta) \tag{3.45}$$

with

$$\xi_P = \frac{m_P^2}{(4\pi f_\pi)^2} \tag{3.46}$$

$$\lambda_P = m_P L \tag{3.47}$$

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$m(n)$	6	12	8	6	24	24	0	12	30	24	24	8	24	48	0	6	48	36	24	24

Table 3.3: Multiplicities for eq. (3.48) as given in [120] for $n \leq 20$.

and

$$\tilde{g}_1(x) = \sum_{n=1}^{\infty} \frac{4m(n)}{\sqrt{n}x} K_1(\sqrt{n}x). \tag{3.48}$$

K_1 is a Bessel function of the second kind. The multiplicities $m(n)$ are calculated in [120] and are replicated in table 3.3.

3.6 Error Analysis and Autocorrelations

Expectation values of observables are calculated using measurements on a number of configurations generated using a Monte Carlo chain. Using a finite number of configurations results in an approximation of the true expectation value. Furthermore, the configurations are not independent of each other if they originate from the same Monte Carlo chain. As a result an error analysis that takes these autocorrelations into account is in order.

Throughout this part the analysis of the statistical errors is done using the autocorrelation function and the Γ-method presented in [122]. The autocorrelation function is defined by

$$\Gamma_{\alpha\beta}(n) = \left\langle \left(a_\alpha^i - A_\alpha\right)\left(a_\beta^{i+n} - A_\beta\right)\right\rangle, \tag{3.49}$$

where the primary observable α is correlated with the primary observable β after n update steps of the Monte Carlo Algorithm. The averages of the observables are given by $A_\alpha = \langle a_\alpha \rangle$. The covariance matrix is given as a sum over the correlation function

$$C_{\alpha\beta} = \frac{1}{N} \sum_{t=-\infty}^{\infty} \Gamma_{\alpha\beta}(t) \tag{3.50}$$

with N being the length of the Monte Carlo chain.

Derived observables F are functions of primary observables A_α

$$F = f(A_\alpha) \tag{3.51}$$

Using the derivatives

$$f_\alpha = \frac{\partial f}{\partial A_\alpha} \tag{3.52}$$

the statistical error of derived observables is calculated as

$$\sigma_F^2 = \sum_{\alpha\beta} f_\alpha f_\beta C_{\alpha\beta}. \tag{3.53}$$

The error analysis hinges on the accurate estimation of the covariance matrix $C_{\alpha\beta}$ and more specifically the sum in eq. (3.50). In practice the sum is truncated to avoid the summation of random fluctuations.

The packages `obs-tools-alpha` and `UWerr` listed in table A.1 handle the error calculation. The software manages the ensembles, the (numerical) calculation of the derivatives in eq. (3.52), the application of the chain rule and works mostly autonomously.

4 | Measurements

In this section we will focus on the calculation of the observables from section 3.3 that are measured on the ensembles described in section 3.1. The majority of the measurements were done prior to this work by colleagues in Mainz [60], Regensburg [61], Wuppertal [63] and Zeuthen [1]. In some cases the different groups use different measurement parameters. They also analyze a different subset of the ensembles and the configurations therein. Furthermore, there exist ensembles with multiple runs that are not replica of each other. They have to be analyzed individually. For these reasons the measurements are evaluated for each data set and each replica run according to table 3.1 individually.

In the following sections we will lay out the details of the measurements of the observables defined in section 3.3. The number of measurements available for each run and each observable is listed in table 4.1.

4.1 Flow Scale

In section 3.3.1 we introduced the Wilson flow. Here, we describe the measurement of the resulting scale t_0 defined implicitly by

$$\left\langle t^2 E(t) \right\rangle \Big|_{t=t_0} = 0.3. \tag{4.1}$$

On the lattice we measure the energy density on every discrete lattice site x. We are using the clover definition of the energy density defined in eq. (3.7). The measurements of the Wilson flow are taken when generating the ensembles or separately using the `ms3` program supplied with the `openQCD` package listed in table A.1.

The energy density at the flow time t is calculated in the following way. We numerically integrate the flow equation given in eq. (3.6) up to the flow time t to get the smoothed gauge fields $V_t(x, \mu)$. We then use the gauge fields at flow time t to calculate the energy density at every point x using the clover definition (eq. (3.7)). This is done for several steps in the flow time $t_i = i\varepsilon,\ i = 1, 2, \cdots$. The size of the steps is $\varepsilon = 0.1$ or $\varepsilon = 0.05$ for some ensembles. The energy density at each of these steps is then averaged over the region of the lattice where boundary effects are small.

The boundary effects of the energy density are studied in [41]. The spacial dimensions use periodic boundary conditions resulting in negligible boundary effects. For the (euclidean) time direction the authors of [41] find that at a distance of

$$M = \left\lceil 12 \cdot \sqrt{t_0^{\mathrm{est}}} \right\rceil \tag{4.2}$$

the effects are negligible. We are using rough estimates t_0^{est} for the determination of this margin. The margins used are listed in third and fourth column of table 4.3.

id	n_{corr}				n_E	n_{rew}	n_{sign}	$n_{\partial m_0 f}$			$n_{\partial m_0 S}$		
	Z	M	R	W				Z	M	W	U	W	Z
H101r000	0	0	1000	0	1007	1007	1007	0	0	0	0	0	1007
H101r001	1009	1009	1000	0	1009	1009	1009	1009	1009	0	0	0	1009
H102r001	1029	1029	1000	0	1029	997	1029	1029	1029	0	0	0	1029
H102r002	1008	1008	1000	0	1008	1008	1008	1008	1008	0	0	0	1008
H105r001	947	1023	996	0	1027	1023	1023	947	1023	0	0	0	947
H105r002	0	1042	1000	0	1042	1042	1042	0	1042	0	0	0	1042
N101r003	0	403	403	0	404	404	404	0	403	0	0	359	0
N101r004	0	239	239	0	240	240	240	0	239	0	0	215	0
N101r005	0	352	352	0	352	352	352	0	352	0	0	352	0
N101r006	0	320	320	0	320	320	320	0	320	0	0	320	0
C101r014	525	2000	2000	0	525	2000	2000	525	2000	0	0	2000	0
B450r000	0	1612	1612	0	1612	1612	1612	0	1612	0	0	1612	0
S400r000	0	872	872	0	872	872	872	0	872	0	0	872	0
S400r001	0	2001	870	0	2001	2001	2001	0	2001	0	0	2000	0
D450r010	0	500	0	0	500	500	500	0	500	0	0	500	0
D452r002	0	1000	0	0	1000	1000	1000	0	1000	0	0	1000	0
N202r001	899	899	884	0	899	899	899	899	899	0	0	0	899
N202r002	0	0	0	1003	1003	1003	0	0	0	1003	0	1003	0
N203r000	756	756	755	0	756	756	756	756	756	0	0	0	756
N203r001	0	787	787	0	787	787	787	0	787	0	0	0	787
N200r000	856	856	856	0	856	856	856	856	856	0	0	0	856
N200r001	856	856	856	0	856	856	856	856	856	0	0	0	856
D200r000	1192	2001	1000	0	1192	2001	2001	1192	2001	0	0	0	1191
E250r001	0	850	393	0	1009	950	1009	0	850	0	0	1009	0
N300r001	506	507	507	0	507	507	507	0	507	0	507	0	0
N300r002	1540	1540	1540	0	1533	1479	1540	1540	1540	0	0	0	1540
N302r001	0	2201	1383	0	2201	2201	2201	0	2201	0	2201	0	0
J303r003	456	1073	630	0	1073	1073	1073	456	1073	0	0	0	517
E300r001	0	1139	0	0	1139	1137	1139	0	1139	0	0	1139	0
J500r004	751	0	751	0	751	751	0	751	0	0	0	0	751
J500r005	0	0	0	0	655	655	0	0	0	0	0	0	0
J501r001	0	0	1497	0	1635	1579	0	0	0	0	1635	0	0
J501r002	0	0	0	0	1142	1142	0	0	0	0	0	0	0

Table 4.1: Number of available measurements for the correlators, the energy density, the reweighting factors, the sign reweighting factors, the derivatives of the correlators and the derivatives of the action. Measurements for the correlators, their derivatives and the derivative of the action exist in different locations: Z: Zeuthen [1, 59], M: Mainz [60], R: Regensburg [61], W: Wuppertal [63], U: Münster [62].

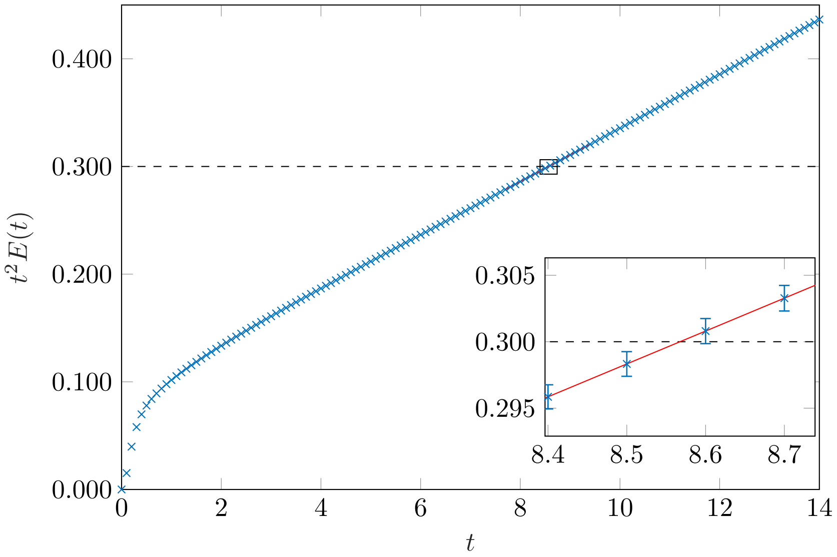


Figure 4.1: Measurements for $t^2E(t)$ as a function of the flow time t on the ensemble N300. The errors in the larger plot have been omitted since they are too small to see. The region around $t^2E(t) \approx 0.3$, defining t_0, is enlarged to visualize the interpolation.

Using $E(t)$ we calculate the dimensionless quantity $t^2E(t)$. It is shown for the N300 ensemble in fig. 4.1. The scale t_0 is calculated from the measurements around $t^2E(t) \approx 0.3$. We are using a quadratic interpolation with the three points closest to the intersection. With $t^\star$ being the point closest to the intersection and $t^{\star,-}$ and $t^{\star,+}$ the points immediately next to it, we can find the coefficients of the polynomial

$$y(t) = c_1t^2 + c_2t + c_3 \tag{4.3}$$

by solving

$$\begin{pmatrix} \left(t^{\star,-}\right)^2 & t^{\star,-} & 1 \\ \left(t^{\star,+}\right)^2 & t^{\star,+} & 1 \\ \left(t^{\star}\right)^2 & t^{\star} & 1 \end{pmatrix} \cdot \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} \left(t^{\star,-}\right)^2 E(t^{\star,-}) \\ \left(t^{\star,+}\right)^2 E(t^{\star,+}) \\ \left(t^{\star}\right)^2 E(t^{\star}) \end{pmatrix}. \tag{4.4}$$

With the coefficients we can then get the intersection with $t^2E(t) = 0.3$ using the quadratic formula

$$t_0 = \frac{-c_2 + \sqrt{c_2^2 - 4c_1\left(c_3 - 0.3\right)}}{2c_1}. \tag{4.5}$$

Through the entire procedure we keep track of the correlations and arrive at the measurements listed in table 4.2.

id	t_0/a^2	am_π	am_k	am_{12}	am_{13}	af_π	af_K	$\sqrt{8\,t_0}f_{\pi K}$
H101	2.8469(51)	0.18306(96)	0.18309(98)	0.009186(71)	0.009186(71)	0.06420(39)	0.06420(39)	0.30637(89)
H102	2.8801(89)	0.15447(81)	0.19170(75)	0.006542(66)	0.010203(64)	0.06113(33)	0.06416(28)	0.3031(12)
H105	2.8865(82)	0.1224(20)	0.2024(15)	0.00396(13)	0.01135(10)	0.0581(18)	0.06458(50)	0.2999(18)
N101	2.8912(34)	0.12193(57)	0.20170(35)	0.004024(27)	0.011345(26)	0.05805(24)	0.06436(18)	0.29941(89)
C101	2.9048(72)	0.09647(77)	0.20580(41)	0.002432(29)	0.011830(34)	0.05502(31)	0.06362(17)	0.29287(86)
B450	3.663(13)	0.16110(56)	0.16108(55)	0.008105(38)	0.008105(38)	0.05660(21)	0.05660(21)	0.30637(77)
S400	3.6910(77)	0.13563(41)	0.17040(40)	0.005687(27)	0.009163(25)	0.05383(34)	0.05701(27)	0.3040(15)
D450	3.6971(54)	0.08361(50)	0.18390(25)	0.002128(26)	0.010792(31)	0.04988(20)	0.05743(11)	0.29864(60)
D452	3.7265(33)	0.05972(63)	0.18645(18)	0.001058(21)	0.011137(14)	0.04776(29)	0.056795(97)	0.29367(68)
N202	5.153(19)	0.13436(30)	0.13431(30)	0.006860(13)	0.006860(13)	0.04860(14)	0.04861(14)	0.31205(45)
N203	5.1466(67)	0.11247(45)	0.14400(38)	0.004741(28)	0.007894(23)	0.04647(22)	0.04916(20)	0.30968(61)
N200	5.1633(62)	0.09239(31)	0.15072(26)	0.003154(13)	0.008646(12)	0.04441(14)	0.04918(15)	0.30583(83)
D200	5.1822(94)	0.06421(63)	0.15578(29)	0.0015404(95)	0.0093822(96)	0.04255(27)	0.04933(11)	0.30308(75)
E250	5.2030(36)	0.04208(23)	0.159291(95)	0.0006396(67)	0.0097557(55)	0.04007(14)	0.04862(12)	0.29531(59)
N300	8.558(41)	0.10632(37)	0.10632(37)	0.0055091(82)	0.0055091(82)	0.03813(20)	0.03813(20)	0.3153(13)
N302	8.526(23)	0.08702(35)	0.11342(35)	0.0037190(88)	0.0063982(86)	0.03643(17)	0.03866(17)	0.3132(13)
J303	8.618(13)	0.06486(21)	0.11979(18)	0.0020448(83)	0.0071963(77)	0.03448(15)	0.03877(20)	0.3101(13)
E300	8.6193(58)	0.04410(39)	0.12377(16)	0.0009253(35)	0.0077285(29)	0.03243(25)	0.03843(15)	0.30253(93)
J500	13.953(40)	0.08121(24)	0.08121(24)	0.0042107(41)	0.0042107(41)	0.02978(13)	0.02978(13)	0.31462(74)
J501	13.994(68)	0.06625(31)	0.08813(31)	0.0027410(41)	0.0049569(38)	0.02865(17)	0.03067(15)	0.31735(91)

Table 4.2: Table of primary analysis results for all considered ensembles.

4.2 Correlators

The measurements of the correlation functions were done by several groups. Here we include measurements from the Mainz (M) [60], Regensburg (R) [61], Wuppertal (W) [63] and Zeuthen (Z) [59] collaborators. In table 4.1 we can see that to a large degree the measurements of different groups were executed on the same configurations. The measurements differ in the position of the source and in the evaluation of the trace described below. Even though the measurements done by different groups are highly correlated, they are combined to increase statistics, exploit the configurations fully and increase the precision of the measurements.

In section 3.3.2, eq. (3.13) we have seen that the correlation function is given by the sum over the spacial volume of the propagators

$$f_X^{rs}(x_0, y_0) = -\frac{a^6}{L^3} \sum_{\vec{x},\vec{y}} \langle \mathrm{tr}\left(\Gamma_X S^r(x,y)\gamma_5 S^s(y,x)\right)\rangle . \tag{4.6}$$

The trace is evaluated using the stochastic techniques described in [64] and the references therein. We get

$$\mathrm{tr}\left(M(x,y)\right) = \left\langle \eta^\dagger_{\alpha,a}(x) M(x,y) \eta_{\alpha,a}(y)\right\rangle_\eta \tag{4.7}$$

where α and a are the Dirac and color indices. The expectation value $\langle\cdot\rangle_\eta$ is calculated as the average over a number of noise sources $\eta_{i,\alpha,a}$. The noise sources themselves used here adhere to following conditions

$$\begin{aligned} \eta_{i,\alpha,a}(x) &\in \mathrm{U}(1) \\ \langle \eta_{\alpha,a}(x)\rangle_\eta &= 0 \\ \left\langle \eta^\dagger_{\alpha,a}(x)\eta_{\beta,b}(y)\right\rangle_\eta &= \delta_{xy}\delta_{\alpha\beta}\delta_{ab}. \end{aligned} \tag{4.8}$$

Using a number N_{src} of random sources η_i the trace is evaluated as the expectation value over the sources and gauge fields

$$\langle \mathrm{tr}\left(\Gamma_X S^r(x,y)\gamma_5 S^s(y,x)\right)\rangle = \left\langle \left\langle \zeta^\dagger_{\beta,b}(y)\xi_{\beta,b}(y)\right\rangle_\eta \right\rangle_U = \left\langle \left\langle \zeta^\dagger_{\beta,b}(y)\xi_{\beta,b}(y)\right\rangle_U \right\rangle_\eta \tag{4.9}$$

using the solution of the Dirac equations

$$\begin{aligned} D^s_{\substack{\alpha,a\\ \beta,b}}(x,y)\gamma_5\xi^s_{\beta,b}(y) &= \eta_{\alpha,a}(x) \quad &\leftrightarrow \quad \xi^s_{\beta,b}(y) &= \gamma_5 S^s_{\substack{\alpha,a\\ \beta,b}}(y,x)\eta_{\alpha,a}(x) \\ D^{r,\dagger}_{\substack{\alpha,a\\ \beta,b}}(x,y)\zeta^r_{\beta,b}(y) &= \Gamma^\dagger_X\eta_{\alpha,a}(x) \quad &\leftrightarrow \quad \zeta^r_{\beta,b}(y) &= S^{r,\dagger}_{\substack{\alpha,a\\ \beta,b}}(y,x)\Gamma^\dagger_X\eta_{\alpha,a}(x) \end{aligned} \tag{4.10}$$

The Dirac equation is solved using the solvers supplied in the `openQCD` package [123]. In particular, the calculation of the correlators is done with the `mesons` package written by Tomasz Korzec. An overview of the software can be found in table A.1.

Concerning the relative and absolute sign of the correlators we adhere to the convention introduced by the ALPHA collaboration, i.e. $f_P(x_0) > 0$ and $f_A(x_0) > 0$ for $x_0 > y_0$.

There are a range of ways to measure different correlation functions. One can vary the type and number of noise sources η_i, the source location y and other algorithmic parameters. Here, most significantly different measurements use different source positions. For that reason it is not possible to average the correlation functions measured on the same ensemble by different groups. Therefore, we evaluate each correlation function on each ensemble and for each measurement. This extends to the pseudoscalar observables we extract from the correlators. The average over measurements on the same ensemble by different groups is only taken when calculating the scale setting observables presented in section 5.1.

4.3 Measurement of Pseudoscalar Observables: Plateau and Fit Ranges

The correlators and accordingly the observables extracted from them are contaminated by excited state and boundary contributions. To avoid these contaminations the temporal extent of the lattice is chosen large enough that a plateau region develops in the center. In this plateau region the contaminations are smaller than the statistical uncertainty and can be neglected. The procedure to extract observables from the correlation functions without the excited state and boundary contaminations consists of the following steps.

1. Characterize the contributions from higher states to the observable in question.
2. Fit these contributions to determine a region where they are smaller than the statistical uncertainty of the data.
3. Calculate the observable in this region neglecting higher contributions.

Details for each of the relevant observables are presented in the following sections.

These plateau and fit ranges are used for the determination of the observable without contaminations from higher states. The derivatives with respect to the quark mass discussed in section 3.4 are calculated and averaged in the same range.

4.3.1 Pseudoscalar Mass

For the plateau and fit ranges of the pseudoscalar mass we distinguish between open and periodic boundary conditions. This is due to the different shape of the correlation functions and different methods used to extract the pseudoscalar mass from them. The methods used are described in section 3.3.3.

Open Boundary Conditions

For open boundary conditions we employ mostly the technique used in [1, 117] and denote explicitly where we deviate from the steps taken therein.

First we characterize the higher contributions to the effective mass. As seen in section 3.3.3, the pseudoscalar correlator f_P exhibits the following asymptotic behavior.

$$f_P^{\mathrm{obc}}(x_0, y_0) = A_1(y_0)e^{-m_{\mathrm{PS}}x_0} + A_2(y_0)e^{-m'x_0} + B_2(y_0)e^{-(E_{2\mathrm{PS}}-m_{\mathrm{PS}})(T-x_0)} + \cdots \tag{4.11}$$

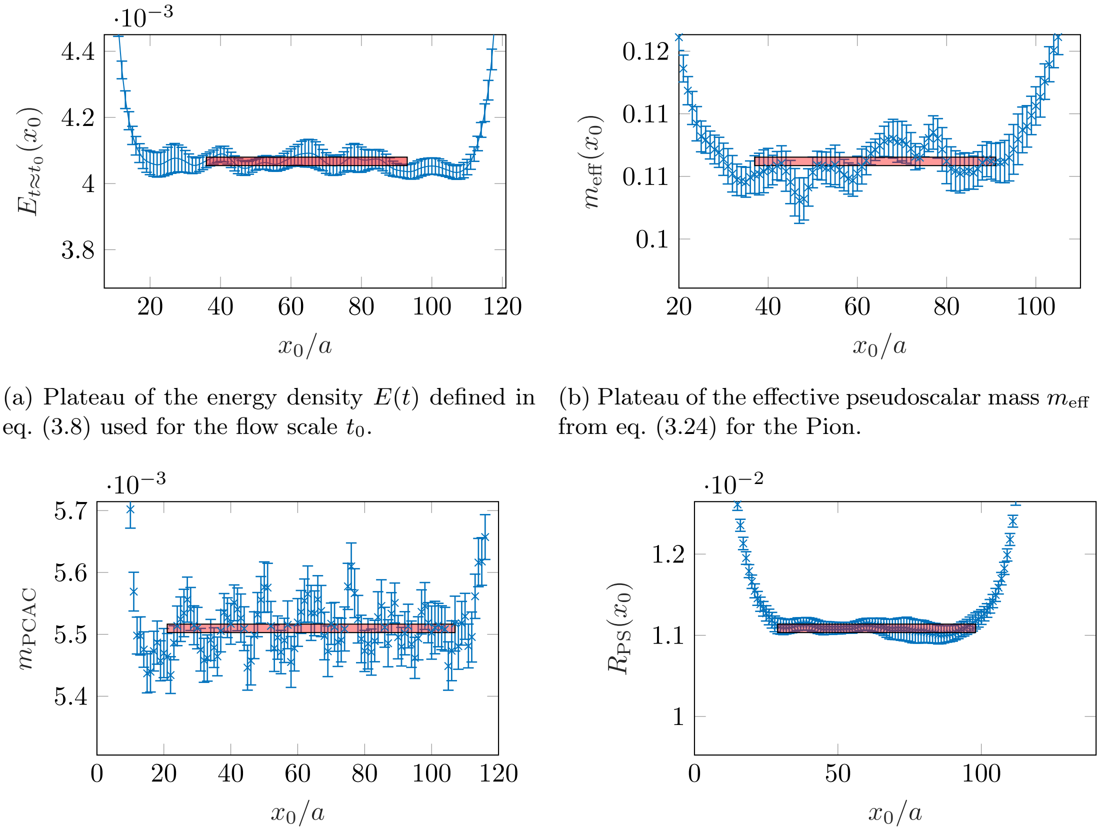
The pseudoscalar mass can be extracted from the pseudoscalar correlator by the following relation as shown in eq. (3.24).

$$a\, m_{\mathrm{eff}}(x_0) = \log\left(\frac{f_P(x_0)}{f_P(x_0+a)}\right) = a\, m_{\mathrm{PS}}\left(1 + c_1 e^{-E_1 x_0} + c_2 e^{-E_{2\mathrm{PS}}(T-x_0)} + \cdots\right) \tag{4.12}$$

A plot of the effective mass for the N300 ensemble is shown in fig. 4.2b. We can see the plateau region in the center as well as the excited state and boundary contributions to the sides.

We now want to characterize the higher contributions in order to remove them from the analysis. If we consider the left and right tail of the excited contributions separately and assume a source close to the boundary, we arrive at

$$f_P(x_0) = \begin{cases} A_1 e^{-m_{\mathrm{PS}}x_0}\left(1 + \tilde{A}_2 e^{-\Delta x_0} + \cdots\right), & (T-x_0)\, m_\pi \quad \gg 1 \\ A_1 e^{-m_{\mathrm{PS}}x_0}\left(1 + \tilde{B}_2 e^{-\Delta' x_0} + \cdots\right), & x_0 m_\pi \qquad \gg 1 \end{cases} \tag{4.13}$$



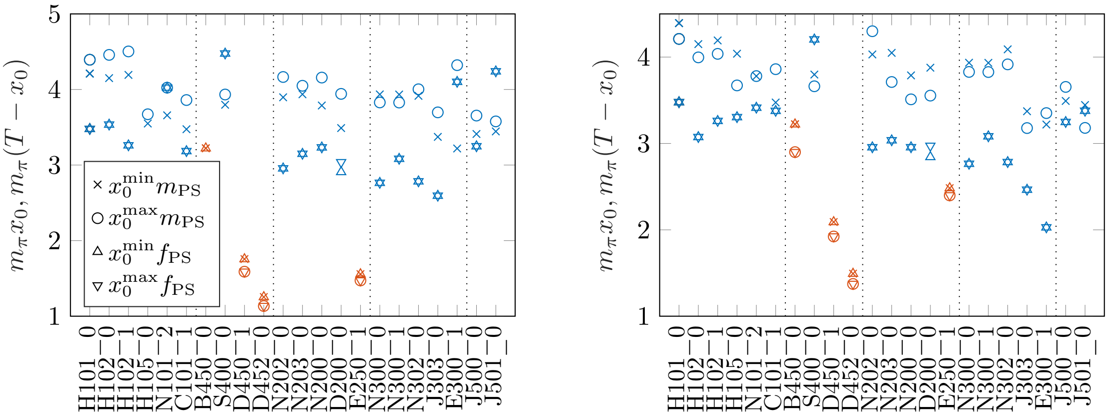
(a) Plateau of the energy density $E(t)$ defined in eq. (3.8) used for the flow scale t_0.

(b) Plateau of the effective pseudoscalar mass m_{eff} from eq. (3.24) for the Pion.

(c) Plateau of the PCAC mass m_{12} from eq. (3.25).

(d) Plateau for the ratio R_{PS} from eq. (3.28) used to calculate the pseudoscalar decay constant for the Pion.

Figure 4.2: Plateaus for the observables extracted from the correlator measurements. The measurements for each timeslice are shown alongside the plateau average. The ensemble shown here uses open boundary conditions in time and the source is located at $x_0 = 1$. The excited state and boundary contributions can be seen to either side of the plateau. The plateau region is determined as presented in sections 4.3.1 to 4.3.3.



(a) Pion.

(b) Kaon.

Figure 4.3: Plateau and fit ranges for the pseudoscalar observables. Ensembles with open boundary conditions are shown in blue, periodic ensembles in red. We can see that, as expected, boundary effects are much smaller for periodic ensembles. The overall size of the boundary effects is similar for most ensembles, which raises the confidence in the procedure.

The mass gaps Δ, Δ' are determined from considering the expansion of the correlator (eq. (3.18)) in eq. (4.11). The pseudoscalar current operator in the correlator selects states by their quantum numbers. The first matrix element in eq. (3.18), $\langle X \,|\, P \,|\, 0\rangle$, results in the exponent $e^{-m_{\mathrm{PS}}}$ that is used to extract the pseudoscalar mass. Here $X = \pi, K$ is the pseudoscalar in question and P is the pseudoscalar current. The next matrix element $\langle X\pi\pi \,|\, P \,|\, \pi\pi\rangle$ determines the mass gap $\Delta = \Delta' \approx 2m_\pi$. If we further assume that the Pion mass is sufficiently small such that these states are the ones with the smallest energy after the ground state, we conclude that the excited state contributions are described by an exponential of the form $e^{-2m_\pi t}$.

Having characterized the higher contributions to the effective mass, we fit the left and right contributions separately with

$$F(x_0) = A \exp\left(-2m_\pi x_0\right) + B. \tag{4.14}$$

Once we have parameterized the excited state contributions we determine the plateau as the region where the excited state contributions are a factor $N_\sigma = 4$ smaller than the error of the plateau average $\sigma(\overline{m_{\mathrm{eff}}})$[1].

$$F(x_0) < \frac{1}{N_\sigma}\sigma(\overline{m_{\mathrm{eff}}}) \tag{4.15}$$

The measurements for the effective mass as well as the plateau region and average are shown in fig. 4.2b for the N300 ensemble as an example. The plateau limits are shown for all ensembles in table 4.3 and fig. 4.3. Plateau averages of the pseudoscalar masses are given in table 4.2.

[1] In contrast to [1], where the plateau is defined using the error of the effective mass on the individual time slice.

id	T	E		m_π		m_K		m_{12}		m_{13}		f_π		f_K	
		←	→	←	→	←	→	←	→	←	→	←	→	←	→
H101	96	21	74	23	71	24	72	18	78	18	78	19	76	19	76
H102	96	21	74	27	66	27	69	18	78	18	78	21	74	21	74
H105	96	21	74	29	65	33	65	18	78	18	78	22	73	27	68
N101	128	21	106	30	94	31	96	18	110	18	110	33	94	28	99
C101	96	21	74	36	55	36	55	18	78	18	78	33	62	35	60
B450	64	23	40	20	45	20	45	14	50	14	50	11	52	11	52
S400	128	23	104	28	98	28	100	18	110	18	110	33	94	31	96
D450	128	23	104	21	108	25	104	14	114	14	114	13	114	13	114
D452	128	23	104	12	116	19	109	14	114	14	114	8	119	16	111
N202	128	28	99	29	96	30	95	19	109	19	109	22	105	22	105
N203	128	28	99	35	91	36	94	19	109	19	109	28	99	27	100
N200	128	28	99	41	82	41	89	19	109	19	109	35	92	32	95
D200	128	28	99	54	61	60	72	19	109	19	109	45	80	44	81
E250	192	28	163	37	156	59	134	16	176	16	176	37	156	59	134
N300	128	36	91	37	91	37	91	21	107	21	107	29	98	29	98
N302	128	36	91	45	81	47	82	21	107	21	107	32	95	32	95
J303	192	36	155	52	134	52	142	21	171	21	171	40	151	38	153
E300	192	36	155	73	93	73	115	21	171	21	171	93	98	46	145
J500	192	45	146	42	146	43	146	24	168	24	168	40	151	40	151
J501	192	45	146	52	137	52	143	24	168	24	168	64	127	51	140

Table 4.3: Plateau/Fit ranges for different observables. Lower (←) and upper (→) ranges are given for each observable and ensemble.

Periodic Boundary Conditions

For periodic boundary conditions the calculation of the effective mass is not suitable. This is due to the correlator shape

$$f_P^{\text{pbc}}(x_0, y_0) = \tilde{A}_1(y_0)\left(e^{-m_{\text{PS}}x_0} + e^{-m_{\text{PS}}(T-x_0)}\right) + \tilde{A}_2(y_0)\left(e^{-\tilde{m}'x_0} + e^{-\tilde{m}'(T-x_0)}\right) + \cdots \tag{4.16}$$

$$\tilde{m}' \approx m_{\text{PS}} + 2m_\pi \tag{4.17}$$

seen in section 3.3.3. The effective mass can only be approximated where one of the exponential terms dominates the other. In this way we lose valuable information in the center of the lattice. For that reason we extract the pseudoscalar mass for periodic ensembles using a direct fit to the correlator.

Similar to the technique for open boundary conditions presented in the previous section we determine the pseudoscalar mass in two steps. Previously, the higher contributions m' were explicitly set. Here they are characterized by the fit parameter M_2. The pseudoscalar mass m_{PS} is described by the parameter M. We first execute a two state fit

$$F(x_0) = A\left(e^{-Mx_0} + e^{-M(T-x_0)}\right) + B\left(e^{-(M+M_2)x_0} + e^{-(M+M_2)(T-x_0)}\right) \tag{4.18}$$

to get the parameters A, B, M and M_2. We then require the second term describing the higher contributions to be a factor $N_\sigma = 4$ smaller than the error of the correlator.

$$B\left(e^{-(M+M_2)x_0} + e^{-(M+M_2)(T-x_0)}\right) < \frac{1}{N_\sigma}\sigma(f_P(x_0)) \tag{4.19}$$

Since the correlator is symmetric around the center, the upper and lower boundaries are the same distance from the source. The fit ranges for the periodic ensembles are found alongside the plateau regions in table 4.3 and visualized in fig. 4.3.

Having found the region where higher corrections can be neglected we perform a second fit with

$$F = A\left(e^{-Mx_0} + e^{-M(T-x_0)}\right) \tag{4.20}$$

to extract the pseudoscalar mass from the fit parameter M.[2] The pseudoscalar masses determined from this second fit are listed in table 4.2.

4.3.2 PCAC Mass

The bare PCAC mass is defined in eq. (3.25). Here, the PCAC mass is only used on the improvement of the pseudoscalar decay constants in eq. (3.33) and the visualization of the mistunings in eq. (5.10). It is not as sensitive to excited state contributions as the pseudoscalar observables. We therefore use an empirical formula to define the margins from the boundaries that determine the plateau region. Similar considerations as in the case of the pseudoscalar masses lead to the following fit ranges which work well for all ensembles.

$$M = \begin{cases} \lceil 7\sqrt{t_0}/a \rceil, & \text{periodic} \\ \lceil 12 + 3\sqrt{t_0}/a \rceil, & \text{open} \end{cases} \tag{4.21}$$

The plateau is then calculated as

$$P = [M, T - M] \tag{4.22}$$

[2] We could have extracted the pseudoscalar mass from the fit in eq. (4.18). For stability reasons of the first fit we decided to perform a second, simpler fit only in the region not contaminated by higher corrections.

and listed in table 4.3. A plateau of the PCAC mass can be seen in fig. 4.2c. Inside the plateau range the individual measurements for the PCAC mass are averaged using their uncertainties as weights. We arrive at the values listed in table 4.2.

4.3.3 Pseudoscalar Decay Constant

The determination of the pseudoscalar decay constant uses different approaches for open and periodic boundary conditions. The methods are described in section 3.3.5. Here we will focus on the determination of the plateau and fit region needed to extract the decay constants. The plateau region is found similarly to the pseudoscalar masses, described in section 4.3.1.

Open Boundary Conditions

In section 3.3.5, eq. (3.28) we have defined the ratio $R_{\mathrm{PS}}(x_0)$. A plot of this ratio for the N300 ensemble can be found in fig. 4.2d. Similar to the pseudoscalar masses we fit the higher contributions with eq. (4.14). The mass gap $2m_\pi$ is used for the Pion as well as for the Kaon decay constants. We then apply a similar criterion to eq. (4.15)

$$F(x_0) < \frac{1}{N_\sigma} \sigma \left(\overline{R_{\mathrm{PS}}} \right) \tag{4.23}$$

to determine the plateau boundaries. Here $\sigma \left(\overline{R_{\mathrm{PS}}} \right)$ is the error of the plateau average of R_{PS} and $N_\sigma = 4$. We apply the fits as well as the criterion in eq. (4.23) to the left and right contributions separately and arrive at the values listed in table 4.3 and shown in fig. 4.3.

Periodic Boundary Conditions

The procedure to extract the pseudoscalar decay constant from the correlators on periodic ensembles is presented in section 3.3.5. Similar to the determination of the pseudoscalar mass for periodic ensembles, the decay constant is extracted from a direct fit to the correlators. We have already calculated the range where higher contributions to the correlators can be neglected in section 4.3.1. We use the same values for the fit range of the pseudoscalar and axial correlators needed to calculate the pseudoscalar decay constant.

5 | Scale Setting

Scale setting involves the precise calculation of one observable, the scale, to act as a reference. Additional lattice measurements are compared to this reference scale to determine their value in physical units. In the lattice community setting the scale is often considered equivalent to the determination of the lattice spacing a.

In chapter 2 we listed a number of different reference scales along with their benefits and disadvantages. Here, we will use the flow scale t_0 introduced in [101, 115] and discussed in sections 3.3.1 and 4.1. The value of the flow scale at the physical point is determined using a combination of the Pion and Kaon decay constants. To set the scale we measure these observables on the ensembles introduced in section 3.1. These ensembles are simulated at finite lattice spacing and (mostly) unphysical quark masses. For this reason, they have to be extrapolated to the continuum and the physical point.

The chiral extrapolation to the physical point runs along the chiral trajectory. This trajectory in the space of the quark masses is chosen such that the improved coupling is constant. Chiral perturbation theory (χPT) is the effective field theory [124] of particles composed of several quarks, in this case Pions and Kaons. It is used here to extrapolate the measurements to the physical point. The chiral trajectory, the mistuning of the ensembles from the trajectory and the correction of the mistuning are discussed in section 5.2. The chiral extrapolation is covered in detail in section 5.3.1.

The continuum limit is executed along the line of constant physics. As the lattice spacing is decreased and the continuum limit is approached, we have to ensure that the underlying physics described by the ensembles does not change. Otherwise, we would end up with an invalid continuum limit. In particular, the dimensionless quantities ϕ_2 and ϕ_4 defined in the next section define the line of constant physics and are kept constant as the lattice spacing is decreased. All relevant lattice observables are $\mathcal{O}\left(a\right)$ improved. We therefore expect lattice artifacts to start at $\mathcal{O}\left(a^2\right)$ and model the behavior toward the continuum accordingly. The continuum limit is presented in section 5.3.2. In practice and to take advantage of the full statistics these extrapolations are executed at the same time. Their individual quality is discussed in section 5.3.3.

Once the chiral and continuum extrapolations are under control, we can determine the scale t_0. For technical reasons the observables used in the extrapolation depend on the scale t_0. We therefore have to find a self consistent solution where the scale that enters the analysis matches the one retrieved at the end. The rough location of this fixed point is known from previous studies [1] and the dependence on the input scale is weak. As a result, the fixed point is easily found. The fixed point procedure is topic of section 5.4.

After the determination of the scale at the physical point, we consider uncertainties of the analysis in section 5.5. The statistical uncertainties are calculated using the fully correlated error analysis presented in [122] and section 3.6. Systematic uncertainties can arise from higher terms in the chiral and continuum extrapolations and uncertainties in the determination of the plateau values. To estimate them we employ a number of different extrapolation techniques. Each extrapolation leads to a slightly different determination of the scale. The systematic uncertainty is estimated from the distribution of these values in section 5.5.

Finally, once the scale is determined, we calculate the lattice spacing of the ensembles used in this analysis in section 5.6.

5.1 Dimensionless Observables

We analyze ensembles at different lattice spacings and quark masses. The inverse coupling β mediates the lattice spacing while the κ parameters determine the bare quark masses. For each inverse coupling β the chiral trajectory starts at the symmetric point where $m_l = m_s$ and extends towards physical quark masses. The trajectory is defined by keeping the sum of quark masses $m_u + m_d + m_s$ constant resulting in a constant inverse coupling β along the chiral trajectory. Details on the chiral trajectory are given in the following section. To study this trajectory systematically it is useful to define the following dimensionless quantities.

$$\phi_2 = 8t_0 m_\pi^2 \qquad \propto m_u + m_d = 2m_l \tag{5.1}$$

$$\phi_4 = 8t_0 \left(m_K^2 + \frac{1}{2} m_\pi^2 \right) \qquad \propto m_u + m_d + m_s = 2m_l + m_s \tag{5.2}$$

The proportionality to the quark masses is derived from leading order chiral perturbation theory [125–127]. These quantities will be used to tune the ensembles to the desired physical parameters and chiral trajectory.

To compute the intermediate scale t_0 we use the following combination of Pion and Kaon decay constants, see sections 3.3.1 and 3.3.5.

$$\sqrt{t_0} f_{\pi K} = \sqrt{t_0} \frac{2}{3} \left(f_K + \frac{1}{2} f_\pi \right) \tag{5.3}$$

The flow scale t_0 can be used to calculate the lattice spacing for each ensemble. The specific combination of Pion and Kaon decay constants is used because of its improved chiral behavior [1, 125, 126]. The chiral extrapolation is discussed in section 5.3.1.

5.2 Chiral Trajectory and Mistuning

In the previous sections we already mentioned the chiral trajectory. It is the path in the space of the quark masses along which we extrapolate towards the point of physical quark masses. Here, we will properly define and illustrate one particular chiral trajectory that has been used in previous studies [1, 128]. It is chosen such that the improved coupling

$$\tilde{g}_0^2 = g_0^2 (1 + a b_g \mathrm{tr}\,(M)) \tag{5.4}$$

along this extrapolation remains constant. The ensembles residing on or close to this chiral trajectory are visualized in fig. 3.1. Their parameters are given in table 3.1. For each inverse coupling $\beta = \frac{6}{g_0^2}$ we start from the symmetric point where the strange and light quark masses are equal. At this point the Pion and Kaon observables are the same. From here we decrease the mass of the light quark and increase the mass of the strange quark while keeping the sum of the quark masses, $\mathrm{tr}\,(M)$, constant. The masses at the symmetric point are tuned such that the chiral trajectory runs roughly through the point of physical quark masses.

In practice there are multiple ways to define the $\mathrm{tr}\,(M) =$ const. trajectory. In leading order chiral perturbation theory the following quantities are all proportional to the sum of the quark masses.

$$\left.\begin{array}{r}\mathrm{tr}\left(M^{\mathrm{bare}}\right) = \sum_f \left(\frac{1}{2\kappa_i} - \frac{1}{2\kappa_{\mathrm{crit}}}\right) \\ \mathrm{tr}\left(M^R\right) \\ \phi_4 = 8t_0\left(m_K^2 + \frac{1}{2}m_\pi^2\right)\end{array}\right\} \propto m_u + m_d + m_s \tag{5.5}$$

Since the renormalized quark masses $\mathrm{tr}\left(M^R\right)$ and ϕ_4 stem from measurements, during the generation of the ensembles only the bare quark masses are available. Accordingly, the parameters κ_i are used to ensure that the ensembles are located on the $\mathrm{tr}\,(M) =$ const. line. Once the ensembles are generated the chiral trajectory can be defined using lattice measurements of ϕ_4. The dimensionless quantity ϕ_4 is a combination of hadronic observables that does not need to be renormalized and is proportional to the sum of quark masses in leading order perturbation theory. It is therefore used to define the chiral trajectory.

The generated ensembles are mistuned from the chiral trajectory for two reasons. The target value of ϕ_4 is not known during the generation of the ensembles. The chiral trajectory used during the generation of the ensembles deviates from the real chiral trajectory due to higher terms in the chiral perturbation theory expansion of ϕ_4. Additionally, the precise value of the physical point is only known once the analysis is finished. This means that the chiral trajectory can miss the physical point. The dependence of ϕ_4 and thus the chiral trajectory on the scale t_0 leads to the fixed point procedure discussed in section 5.4.

These mistunings can be seen in the measurements of ϕ_4 shown in fig. 5.1a and need to be addressed in order to get on the line of constant physics. The observable $\phi_2 \propto m_l$ gives a sense of the decreasing mass of the light quarks, while $\phi_4 \propto 2m_l + m_s$ is a proxy for the sum of the quark masses. The dashed line indicates the physical value of ϕ_4 which is the target for the correct trajectory. The measured values for ϕ_4 show the mistuning of the ensembles.

To correct this mistuning we measure the derivatives of all observables with respect to the bare quark masses. Details of these measurements are found in section 3.4. If we assume that the mistuning is small, we can shift an observable X to the correct trajectory using a first order Taylor expansion in the bare quark masses.

$$X(m_u', m_d', m_s') = X(m_u, m_d, m_s) + \delta m \sum_i n_i \frac{\partial}{\partial m_i} X(m_u, m_d, m_s) + \mathcal{O}\left(\delta m^2\right) \tag{5.6}$$

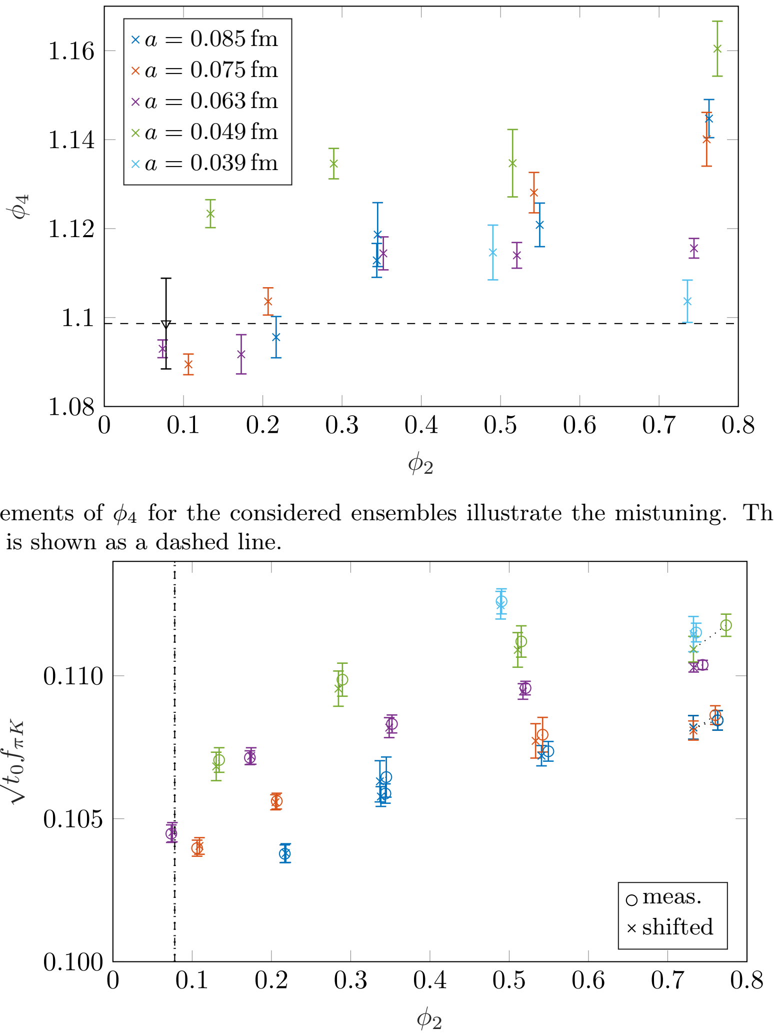
We have separated the quark mass shift into the magnitude $\delta m = m_i' - m_i$ and the normalized direction $\vec{n} = (n_u, n_d, n_s)$. The optimal direction for this shift is subject of section 5.2.1. If we have set a target direction $\vec{n}$, we can calculate the magnitude of the shift from the mistuning of ϕ_4.

$$\delta m = \frac{\phi_4^{\mathrm{phys}} - \phi_4}{\sum_i n_i \frac{d}{dm_i}\phi_4} \tag{5.7}$$

We can now shift the scale setting observable $\sqrt{t_0} f_{\pi K}$ to the chosen trajectory.

Since in the end the shift is determined by the difference $\delta\phi_4 = \phi_4^{\mathrm{phys}} - \phi_4$, it is convenient to express the derivatives with respect to ϕ_4 as well. This can easily be done using the quark mass derivative of ϕ_4 calculated earlier.

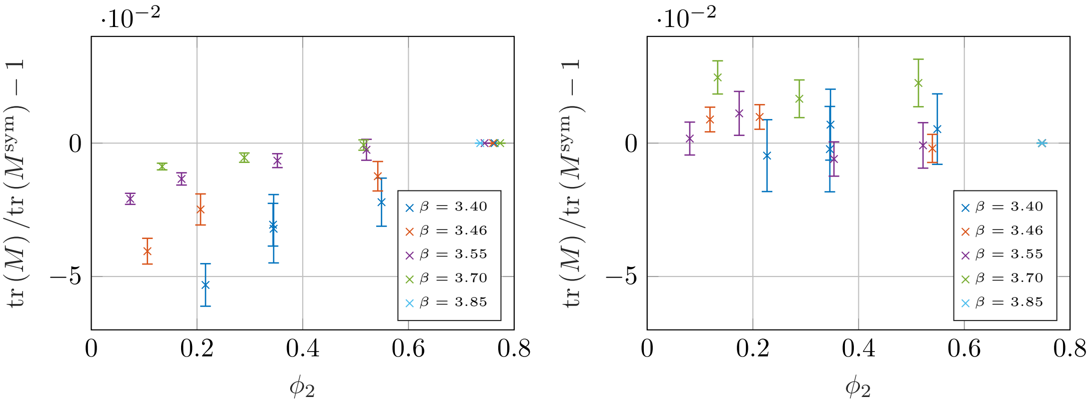
$$\frac{dX}{d\phi_4} = \frac{\sum_i n_i \frac{\partial}{\partial m_i} X}{\sum_i n_i \frac{\partial}{\partial m_i}\phi_4} \tag{5.8}$$



(a) Measurements of ϕ_4 for the considered ensembles illustrate the mistuning. The physical value of ϕ_4 is shown as a dashed line.

(b) Measurements for $\sqrt{t_0}f_{\pi K}$ and the corresponding values shifted to $\phi_4 = \text{const.}$ along the $\vec{n} = (0,0,1)^T$ direction.

Figure 5.1: Effects of the mistuning and the quark mass shifts. The left plot illustrates how the measured sum of quark masses ϕ_4 fluctuates. The right plot shows the original measurements for $f_{\pi K}$ as well as the measurements shifted to the correct trajectory.



(a) Original $\mathrm{tr}\left(M^{\mathrm{bare}}\right) = \mathrm{const.}$ trajectory used during the ensemble generation.

(b) Trajectory defined by $\phi_4 = \mathrm{const.}$.

Figure 5.2: Measurements of the deviation from the $\mathrm{tr}\,(M) = \mathrm{const.}$ trajectory. The two trajectories shown differ in the higher terms of the expansion. The relative deviation as well as the discretization effects are smaller for the $\phi_4 = \mathrm{const.}$ trajectory.

The shifts are then calculated according to

$$X(\phi_4') = X(\phi_4) + \delta\phi_4 \frac{\partial X}{\partial \phi_4} + \mathcal{O}\left(\left(\frac{\delta\phi_4}{\phi_4}\right)^2\right) \tag{5.9}$$

To study the effects of the shift to the correct chiral trajectory we consider measurements of the quark masses. Using measurements of the PCAC masses m_{12} and m_{13} defined in section 3.3.4 and discussed in section 4.3.2 we calculate the sum of the quark masses

$$\mathrm{tr}\left(M^{\mathrm{bare}}\right) = \frac{1}{2} m_{12}^{\mathrm{bare}} + m_{13}^{\mathrm{bare}}. \tag{5.10}$$

The sum of the quark masses

$$\frac{\mathrm{tr}\,(M)}{\mathrm{tr}\,(M^{\mathrm{sym}})} = 1 + \frac{1}{3} b_R a \,\mathrm{tr}\,(M^{\mathrm{sym}}) \left(\frac{3m_{12}}{\mathrm{tr}\,(M^{\mathrm{sym}})} - 1\right)^2 + \mathcal{O}\left(a^2\right) \tag{5.11}$$

is constant up discretization effects [117]. It is precisely these discretization effects that can be seen in fig. 5.2a as a systematic deviation from zero. As expected these discretization effects decrease as the lattice spacing gets smaller.

When tuning to $\phi_4 = \mathrm{const.}$ the situation changes. In addition to discretization effects we now get mistunings from χPT since $\phi_4 \propto \mathrm{tr}\,(M) + \mathcal{O}\left(m^2\right)$ only matches up to higher orders in the quark masses. In fig. 5.2b we show the mistunings after the measurements have been shifted to the $\phi_4 = \mathrm{const.}$ trajectory. We notice that these mistunings are not systematic and largely independent of the lattice spacing and conclude that they originate predominantly from χPT and discretization effects play a minor role in this tuning.

5.2.1 Direction of the Derivative

In the previous section we discussed how to shift the measurements to any chiral trajectory close to the simulated points. Both for the shift in the quark mass shown in eq. (5.6) and in the

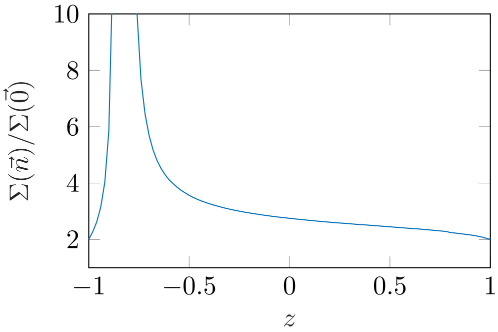


Figure 5.3: Normalized cumulative error defined in eq. (5.13) for different directions according to eq. (5.12). The minimum is located at $z = 1$ corresponding to the direction $\vec{n} = (0, 0, 1)^T$, i.e. shifts only in δm_s.

definition of the derivatives with respect to ϕ_4 we have left the direction $\vec{n}$ unspecified. The vector $\vec{n}$ is a three vector in the space of the quark masses (m_u, m_d, m_s), that is assumed to be normalized. Several shift directions stand out. The direction $\vec{n} = (1, 1, 1)/\sqrt{3}$ preserves the symmetry. This is especially important for the ensembles at the symmetric point. Additionally, we want to keep degenerate light quark masses $m_u = m_d = m_l$. This leads us to the general direction

$$\vec{n} = \left(\sqrt{\frac{1-z^2}{2}}, \sqrt{\frac{1-z^2}{2}}, z\right)^T . \tag{5.12}$$

Due to the uncertainties in the determination of the quark mass derivatives, on average shifting the observables increases their error.[1] This can be seen for example in fig. 5.1b. The goal is to find a shift direction with the restriction given in eq. (5.12) that results in the smallest increase of the error of the scale setting quantity $\sqrt{t_0} f_{\pi K}$. For that reason we define the cumulative error $\Sigma(\vec{n})$, i.e. the sum of the errors σ_i on all ensembles, as a function of the shift direction $\vec{n}$.

$$\Sigma(\vec{n}) = \sum_i \sigma_i \left(\sqrt{t_0} f_{\pi K} \big|_{\vec{m}=\vec{m}_0+\delta m \vec{n}} \right) \tag{5.13}$$

We can now optimize the direction with the cumulative error as a cost function. Figure 5.3 shows that the error depends drastically on the shift direction. We found the optimal direction to be $\vec{n}_{\text{min}} = (0, 0, 1)^T$. We therefore shift the symmetric ensembles in the $\vec{n} = (1, 1, 1)^T$ direction to preserve the symmetry between the light and strange quarks and the other ensembles in the $\vec{n} = (0, 0, 1)^T$ direction to minimize the uncertainties.

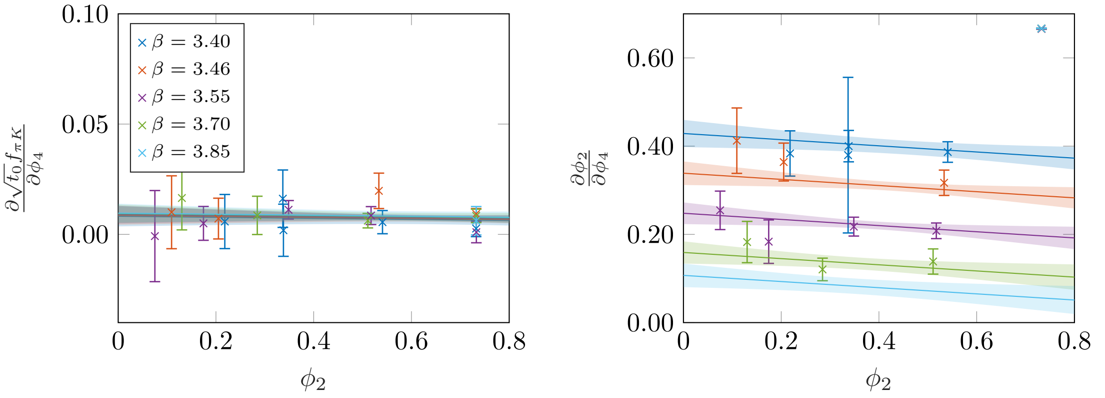
5.2.2 Modeling the Derivatives

Using the derivatives with respect to ϕ_4 introduced in eqs. (5.8) and (5.9), we can get a combined description of the ensembles. This way, ensembles with large uncertainties for the measurements of the quark mass derivatives can benefit from precise measurements on other ensembles. Modeling the derivatives also allows us to extrapolate the measurements to ensembles where measurements of the quark mass derivatives are not available. As a result we can include the ensemble `J501` in the analysis that is important for the stability of the continuum limit.

[1] For observables that are strongly correlated to ϕ_4, such as the Pion and Kaon masses, the uncertainty can be decreased by the shift to a given, fixed value of ϕ_4^{phys}. This is not the case for the scale setting quantity $\sqrt{t_0} f_{\pi K}$.

	A	B	C	χ^2/dof
$\frac{\partial t_0}{\partial \phi_4}$	$-0.48(8)$	$-0.06(9)$	$0.0(3)$	0.9
$\frac{\partial \phi_2}{\partial \phi_4}$	$0.02(3)$	$-0.07(6)$	$1.15(10)$	0.6
$\frac{\partial \sqrt{t_0} f_\pi}{\partial \phi_4}$	$0.008(5)$	$-0.008(7)$	$0.005(14)$	0.6
$\frac{\partial \sqrt{t_0} f_K}{\partial \phi_4}$	$0.016(4)$	$0.006(5)$	$-0.024(12)$	0.7
$\frac{\partial \sqrt{t_0} f_{\pi K}}{\partial \phi_4}$	$0.010(5)$	$-0.002(6)$	$-0.005(13)$	0.6

Table 5.1: Fit parameters of the quark mass derivatives according to eq. (5.14).



(a) Derivative of the scale setting quantity $\sqrt{t_0} f_{\pi K}$.

(b) Derivative of the dimensionless observable ϕ_2. The symmetric points at $\frac{\partial \phi_2^{\mathrm{sym}}}{\partial \phi_4^{\mathrm{sym}}} = \frac{2}{3}$ are excluded from the fit.

Figure 5.4: Measurements and fits of the derivatives of various observables with respect to ϕ_4. The parameters of these fits is given in table 5.1.

We fit the following ansatz to the measurements of the different derivatives to get a function of the lattice spacing and ϕ_2.

$$F(\phi_2) = A + B\phi_2 + C\frac{a^2}{t_0^{\mathrm{sym}}} \tag{5.14}$$

Here A, B and C are parameters determined by the fit. Their values are found in table 5.1. Figure 5.4 shows measurements of the derivatives along with the fits. We present fits for the scale setting quantity $\sqrt{t_0} f_{\pi K}$ as well as ϕ_2. In both cases the dependence on ϕ_2 is weak. Additionally, $\frac{\partial \sqrt{t_0} f_{\pi K}}{\partial \phi_4}$ does not significantly depend on the lattice spacing a.

The fit for $\frac{\partial \phi_2}{\partial \phi_4}$ uses only the non-symmetric ensembles. At the symmetric point we have the exact relation $\phi_2^{\mathrm{sym}} = \frac{2}{3}\phi_4^{\mathrm{sym}}$. The derivative is therefore fixed to $\frac{\partial \phi_2^{\mathrm{sym}}}{\partial \phi_4^{\mathrm{sym}}} = \frac{2}{3}$. This point can also be seen in fig. 5.4b. To preserve this relation we shift the symmetric ensembles in the $\vec{n} = (1,1,1)^T/\sqrt{3}$ direction.

For the same symmetry reasons the measurements of $\sqrt{t_0} f_{\pi K}$ at the symmetric point are also shifted in the $\vec{n} = (1,1,1)^T/\sqrt{3}$ direction. For the symmetric ensembles we use the individual measurements of the derivatives $\frac{\partial\sqrt{t_0}f_{\pi K}}{\partial\phi_4}$. For the remaining ensembles the derivatives are determined using the fit shown in fig. 5.4a. In this fit we also include the symmetric ensembles shifted in the $\vec{n} = (0,0,1)^T$ direction.

In the following we will always use the derivatives from the fits for the non-symmetric ensembles and individual measurements for the symmetric ensembles.

5.3 Extrapolating Lattice Measurements

In order to set the scale we want to compare lattice measurements to physical measurements. Lattice measurements are, however, subject to artifacts caused by the lattice spacing a. Additionally, the measurements are done on unphysical quark masses as discussed in section 3.1. To match the lattice measurements to the physical ones we have to attend and eliminate these artifacts. In practice this means extrapolating the measurements towards the physical point and the continuum. The extrapolation towards the physical point is called chiral extrapolation and is mediated by $\phi_2 \to \phi_2^{\text{phys}}$. It is discussed in section 5.3.1. The continuum extrapolation $a \to 0$ is topic of section 5.3.2. Each extrapolation works by characterizing and modeling the respective behavior and fitting the corresponding ansatz to the measurements. In this case the two extrapolation are executed at the same time using a combined fit. In the following sections we will consider the two extrapolations individually before showing results of the combined fit.

5.3.1 Chiral Extrapolation

The chiral extrapolation is the extrapolation from ensembles with unphysical bare quark masses to the physical point. An overview of the ensembles and their trajectory towards the physical point is shown in fig. 3.1. The physical point is defined by the physical masses of the Pion and Kaon masses. The physical masses can be directly[2] translated into the dimensionless variables $(\phi_2^{\text{phys}}, \phi_4^{\text{phys}})$. The shifting of the measurements discussed in section 5.2 ensures that the chiral trajectory resides on the physical value ϕ_4^{phys}. The chiral extrapolation works to characterize the behavior of the scale setting quantity $\sqrt{t_0}f_{\pi K}(\phi_2)$ as $\phi_2 \to \phi_2^{\text{phys}}$ approaches its physical value.

We use three different techniques to extrapolate to the physical point. This is done to estimate the systematic errors in section 5.5. The extrapolations shown here correspond to the continuum behavior. Lattice effects are discussed in section 5.3.2.

Taylor Expansion

The first and most basic technique is not based on chiral perturbation theory, but instead consists of a Taylor expansion around the symmetric point at ϕ_2^{sym}. As shown in [129] the linear (and cubic) term does not contribute. The Taylor expansion and fit formula is given by

$$F^{\text{cont}}_{\text{Taylor}}(\phi_2, P_1, P_2) = P_1 + P_2(\phi_2 - \phi_2^{\text{sym}})^2. \tag{5.15}$$

The parameters P_1 and P_2 are determined by the fit. The symmetric ϕ_2 is calculated from $\phi_2^{\text{sym}} = \frac{2}{3}\phi_4^{\text{sym}} \equiv \frac{2}{3}\phi_4$.

In addition to the quadratic Taylor expansion given in eq. (5.15) we also use the 4th order Taylor expansion.

$$F^{\text{cont}}_{\text{Taylor(4)}}(\phi_2, P_1, P_2, P_3) = P_1 + P_2(\phi_2 - \phi_2^{\text{sym}})^2 + P_3\,(\phi_2 - \phi_2^{\text{sym}})^4 \tag{5.16}$$

[2] To translate the pseudoscalar masses into the dimensionless variables ϕ_2, ϕ_4 one would need to know the scale t_0 beforehand. How to choose this scale at this stage is discussed later in section 5.4.

SU(3) **Chiral Perturbation Theory**

Chiral perturbation theory (χPT) relates hadronic quantities such as the pseudoscalar masses and decay constants to quark masses which are parameters of the lattice theory. The theory uses a number of low energy constants B_0, L_x. In next to leading order (NLO) χPT the flow scale t_0 is constant [130]. The leading order (masses) and next-to-leading order (decay constants) SU(3) χPT expansion of the hadronic observables are given in [125, 126].

$$m_\pi^2 = B_0\left(m_u + m_d\right) \tag{5.17}$$

$$m_K^2 = B_0\left(m_u + m_s\right) \tag{5.18}$$

$$f_\pi = f\left[1 + \frac{16B_0L_5}{f^2}m_l + \frac{8B_0L_4}{f^2}\left(2m_l + m_s\right) - 2L\left(m_\pi^2\right) - L\left(m_K^2\right)\right] \tag{5.19}$$

$$f_K = f\left[1 + \frac{8B_0L_5}{f^2}\left(m_l + m_s\right) + \frac{16B_0L_4}{f^2}\left(2m_l + m_s\right) - \frac{3}{4}L\left(m_\pi^2\right) - \frac{3}{2}L\left(m_K^2\right) - \frac{3}{4}L\left(m_\eta^2\right)\right] \tag{5.20}$$

The logarithms $L\left(m^2\right)$ are defined as

$$L\left(m^2\right) = \frac{m^2}{4\pi f^2}\log\frac{m^2}{4\pi f^2}. \tag{5.21}$$

Combining eqs. (5.19) and (5.20) according to $f_{\pi K}$ defined in eq. (5.3) we arrive at the fit formula

$$F^{\text{cont}}_{\text{SU(3)}\chi\text{PT}}(\phi_2, P_4, P_5) = P_4\left[1 - \frac{7}{6}L_\pi(\phi_2, P_4) - \frac{4}{3}L_K(\phi_2, P_4) - \frac{1}{2}L_\eta(\phi_2, P_4) + P_5\right]. \tag{5.22}$$

We introduced the fit parameters P_4 and P_5 that are defined in terms of the SU(3) χPT low energy constants f, B_0, L_4 and L_5

$$\begin{aligned} P_4 &= \sqrt{t_0}f \\ P_5 &= \frac{8B_0\text{tr}\left(M\right)}{3f^2}\left(2L_5 + 5L_4\right). \end{aligned} \tag{5.23}$$

The function in eq. (5.22) is fitted to the measurements of $\sqrt{t_0}f_{\pi K}$. The logarithms from eq. (5.21) can be expressed in terms of ϕ_2 and ϕ_4 using their respective definitions

$$L_\pi(\phi_2, \sqrt{t_0}f) = \frac{\phi_2}{(4\pi)^2 8\, t_0\, f^2}\log\left(\frac{\phi_2}{(4\pi)^2 8\, t_0\, f^2}\right) \tag{5.24}$$

$$L_K(\phi_2, \sqrt{t_0}f) = \frac{\phi_4 - \frac{1}{2}\phi_2}{(4\pi)^2 8\, t_0\, f^2}\log\left(\frac{\phi_4 - \frac{1}{2}\phi_2}{(4\pi)^2 8\, t_0\, f^2}\right) \tag{5.25}$$

$$L_\eta(\phi_2, \sqrt{t_0}f) = \frac{\phi_4 - \frac{3}{4}\phi_2}{(4\pi)^2 6\, t_0\, f^2}\log\left(\frac{\phi_4 - \frac{3}{4}\phi_2}{(4\pi)^2 6\, t_0\, f^2}\right). \tag{5.26}$$

The last logarithm is transformed using the η mass

$$m_\eta^2 = \frac{1}{6t_0}\left(\phi_4 - \frac{3}{4}\phi_2\right). \tag{5.27}$$

In [1] the chiral extrapolation was done using the ratio $F_{\text{SU(3)}\chi\text{PT}}(\phi_2)/F_{\text{SU(3)}\chi\text{PT}}(\phi_2^{\text{sym}})$. Normalizing by the symmetric ensembles eliminates the fit parameter P_5. However, it also emphasizes the symmetric ensembles. For that reason we chose to implement the additional fit parameter P_5 and treat the symmetric ensembles in line with the others.

SU(2) **Chiral Perturbation Theory**

Exact SU(3) symmetry is only present in the limit of vanishing quark masses. The heavier strange quark breaks strongly this symmetry. It is therefore worthwhile to consider the remaining SU(2) symmetry between the light quarks. Chiral perturbation theory in SU(2) is discussed in [131]. The pseudoscalar decay constants are given in terms of the low energy constants P_6 to P_9 and the logarithm in eq. (5.24).

$$F^{\pi,\text{cont}}_{\text{SU(2)}\chi\text{PT}}(\phi_2, P_6, P_7) = P_6\phi_2 + P_7\left[1 + 2L_\pi(\phi_2, P_7)\right] \tag{5.28}$$

$$F^{K,\text{cont}}_{\text{SU(2)}\chi\text{PT}}(\phi_2, P_7, P_8, P_9) = P_8\phi_2 + P_9\left[1 + \frac{3}{4}L_\pi(\phi_2, P_7)\right] \tag{5.29}$$

The parameters P_6 to P_9 are related to the SU(2) χPT low energy constants defined in [131]. Note that the Pion parameter P_7 also occurs in the logarithm for the Kaon decay constant. For that reason we fit the two observables at the same time with eqs. (5.28) and (5.29) respectively. From the two individual fits we can then reconstruct $f_{\pi K}$ to set the scale.

5.3.2 Continuum Extrapolation

So far we have discussed how to extrapolate the measurements of $\sqrt{t_0}f_{\pi K}$ to the physical point given by $(\phi_2^{\text{phys}}, \phi_4^{\text{phys}})$. We have yet to characterize the lattice artifacts to extrapolate the measurements to the continuum along the line of constant physics defined by $\phi_2 = \text{const.}$. All observables considered here have been improved such that order $\mathcal{O}(a)$ effects vanish (see chapter 1 for details). The leading term is therefore of order $\mathcal{O}\left(a^2\right)$. The degree to which higher terms can be neglected is discussed in sections 5.3.3 and 5.5. To account for the effects of the lattice spacing a, we multiply the chiral extrapolation function by an a^2 term.

$$F_\chi(\phi_2, \cdots, P_{10}) = F_\chi^{\text{cont}}(\phi_2, \cdots) \cdot \left(1 + P_{10}\frac{a^2}{t_0}\right) \tag{5.30}$$

Here $F_\chi^{\text{cont}}(\phi_2, \ldots)$ is any of the chiral extrapolations presented in section 5.3.1. The ellipsis corresponds to the parameters of the respective chiral extrapolation. The parameter P_{10} along with the parameters of the chiral extrapolation are determined by the combined fit.

Additionally, we also added a term proportional to $a^2 m_\pi^2$ to model the dependence on the lattice spacing a.

$$F_\chi(\phi_2, \cdots, P_{10}, P_{11}) = F_\chi^{\text{cont}}(\phi_2, \cdots) \cdot \left(1 + P_{10}\frac{a^2}{t_0} + P_{11}a^2 m_\pi^2\right) \tag{5.31}$$

This term provides a ϕ_2 dependence of the continuum extrapolation but adds another parameter to the fit. The different extrapolation methods will be discussed in sections 5.3.3 and 5.5.

5.3.3 Combined Extrapolation Results

In this section we will discuss the combined chiral and continuum extrapolation introduced in sections 5.3.1 and 5.3.2. Because the physical point depends on the results of these fits we use a reference point at $(\phi_2, \phi_4) = (0.075, 1.12)$ to compare the different extrapolations. First the measurements are shifted to $\phi_4 = 1.12$ using the method described in section 5.2.2. This value is chosen as it is close to the average of all ϕ_4 measurements (see fig. 5.1a). Then the combined chiral and continuum extrapolation is executed. As an example, the SU(3) chiral perturbation theory extrapolation defined in eq. (5.22) together with the term describing the

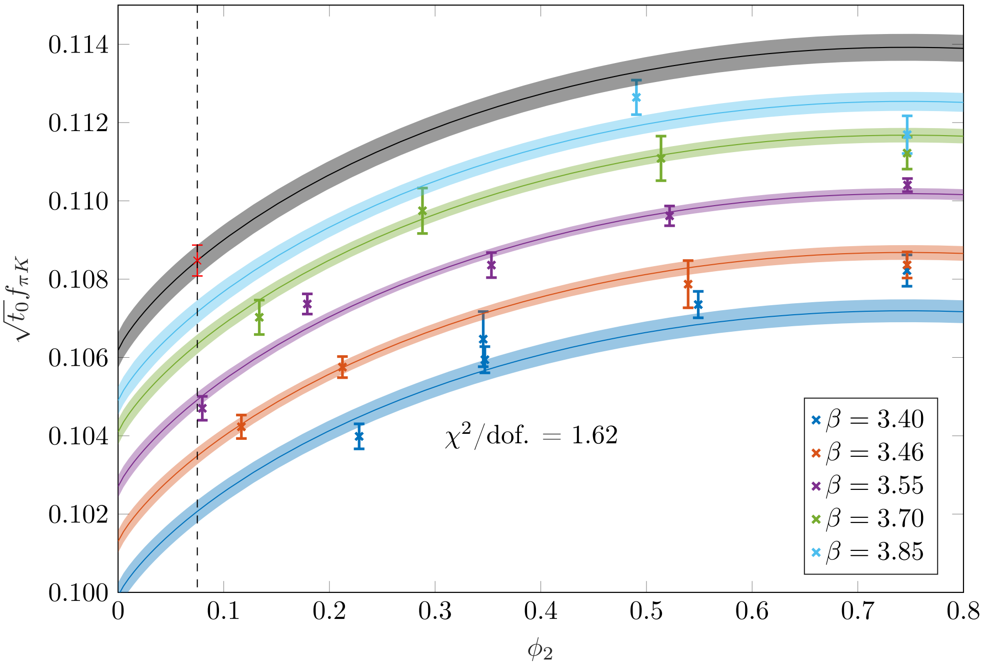
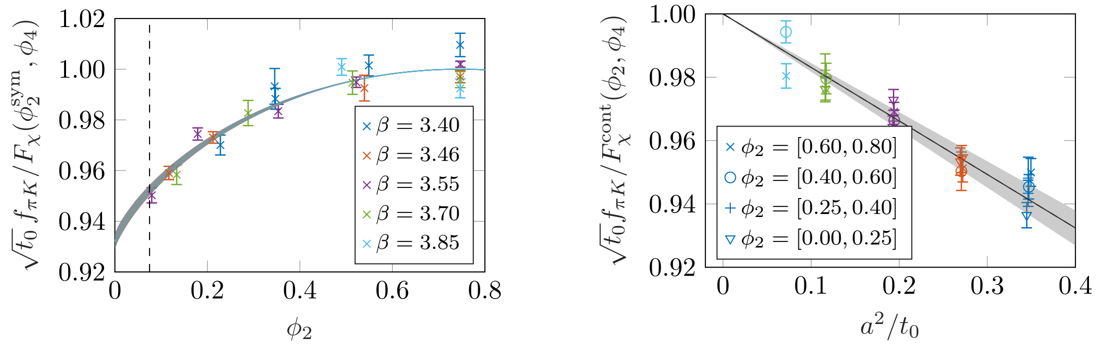


Figure 5.5: Measurements for the scale setting quantity $\sqrt{t_0} f_{\pi K}$ alongside the combined chiral and continuum extrapolation. Different lattice spacings are shown in different colors with the continuum curve shown in gray. Shown here is the SU(3) chiral perturbation theory extrapolation defined in eqs. (5.22) and (5.30). During the fit we omitted the coarsest lattice spacing corresponding to a $\beta > 3.4$ cut.



(a) Ratio $R_{\rm chiral}$ defined in eq. (5.32) illustrating the chiral extrapolation.

(b) Ratio $R_{\rm cont}$ defined in eq. (5.33) illustrating the continuum extrapolation.

Figure 5.6: Ratios of the combined fit function presented in fig. 5.5. The two ratios defined in eqs. (5.32) and (5.33) isolate the chiral and continuum components of the extrapolation. Here they are shown together with the normalized measurements for the scale setting quantity $\sqrt{t_0}f_{\pi K}$. The method shown here consists of the SU(3) χPT extrapolation and a $\beta > 3.4$ cut to the data.

lattice artifacts in eq. (5.30) is shown in fig. 5.5. Different lattice spacings are indicated by different colors. The continuum limit extends upwards towards the gray line which represents the chiral extrapolation in the continuum. The chiral extrapolation in ϕ_2 runs toward the reference point $\phi_2 = 0.075$. The value of the continuum line at the reference point is then compared between different extrapolation techniques described in section 5.3.1. In the extrapolation shown in fig. 5.5 we applied the cut $\beta > 3.4$, discarding the coarsest lattices with $a = 0.085\,\mathrm{fm}$. The remaining data points are well described by the fit. This is reflected by the value of the residue $\chi^2/\mathrm{dof.} = 1.62$ (see table 5.2).

To better judge the quality of the individual components of the fit we have isolated the chiral and continuum parts in fig. 5.6. First the measurements are divided by their respective fit function evaluated at the symmetric point $\phi_2^{\rm sym}$. This isolates the chiral component of the extrapolation function since the parts describing the lattice artifacts cancel. We use the extrapolations discussed in sections 5.3.1 and 5.3.2 to define the ratio

$$R_{\rm chiral} = \frac{F_\chi(\phi_2, \cdots)}{F_\chi(\phi_2^{\rm sym}, \cdots)} = \frac{F_\chi^{\rm cont}(\phi_2, \cdots)}{F_\chi^{\rm cont}(\phi_2^{\rm sym}, \cdots)}. \tag{5.32}$$

The ellipsis corresponds to the parameters of the respective chiral extrapolation. The chiral ratio is shown in fig. 5.6a together with the measurements. Similarly, the ratio

$$R_{\rm cont} = \frac{F_\chi(\phi_2, \cdots)}{F_\chi^{\rm cont}(\phi_2, \cdots)} = 1 + P_{10}\frac{a^2}{t_0} \tag{5.33}$$

isolates the term that describes the continuum extrapolation. This second ratio is shown in fig. 5.6b alongside the measurements. Note that the lines in fig. 5.6 do not indicate a separate fit of the ratios but is instead an evaluation of the global fit shown in fig. 5.5.

For the chiral ratio shown in fig. 5.6a we can see the data points fluctuating around the lines. No systematic deviation is visible. We conclude that the chiral behavior is well described by the SU(3) ansatz presented in section 5.3.1. The isolated continuum behavior in fig. 5.6b is well described by the a^2 term defined in section 5.3.2 and eq. (5.30). Only the coarsest lattice spacing indicated by the dark blue points to the right show a slight deviation from the a^2 line. Rather than implementing an additional continuum term we choose to omit the coarsest lattices from the analysis by employing a $\beta > 3.4$ cut to the data points.

To compare different chiral extrapolations we evaluate the extrapolation function at the reference point $\phi_2 = 0.075$. The results of the extrapolation techniques described in section 5.3.1 are shown in table 5.2. We present the type of extrapolation as well as the cut used. To compare the quality of the fit we list the value of $\chi^2/\text{dof}.$ and the value of the scale setting quantity $\sqrt{t_0}f$ at the reference point and in the continuum. For the Taylor and SU(3) extrapolations $f = f_{\pi K}$ is used. The SU(2) extrapolations uses $f = f_\pi$ to set the scale t_0.

The different extrapolation techniques and cuts are very stable. The scale setting quantity $\sqrt{t_0}f$ fluctuates from $\sqrt{t_0}f = 0.1074$ to $\sqrt{t_0}f = 0.1090$ across all methods and cuts. This change is ±1% of the average value and of the same magnitude as the average statistical uncertainty. This small fluctuation is due to the precise measurements close to the physical point.

The quality of the fit is indicated by the χ^2/dof. The values lie between roughly 1 and 2.6. Omitting the symmetric ensembles the farthest away from the reference point results in a better fit. The quality of the fits will be more closely examined in section 5.5 after the physical point is determined.

The quality of the 4^{th} order Taylor expansion fit is only slightly better than that of the quadratic expansion. Using the SU(3) chiral perturbation function fit we model two different continuum functions listed in eqs. (5.30) and (5.31). With the addition of the $a^2m_\pi^2$ term the value for $\sqrt{t_0}f$ stays mostly within one standard deviation while the quality of the fit is not improved in all cases. The results for the SU(3) and SU(2) χPT functions are very similar. The fit quality is better for the SU(2) χPT function.

The different extrapolations are used in section 5.5 to estimate the systematic error at the physical point.

5.4 Finding the Physical Point

In the previous section we have shifted the measurements to the target value for ϕ_4, extrapolated the scale setting quantity $\sqrt{t_0}f$ to the continuum and evaluated the function at the target ϕ_2. We can now extract the flow scale t_0 by comparing the lattice determination of $\sqrt{t_0}f$ to physical measurements of the decay constants. We use the measurements reported in [9] where the effects of QCD and the degenerate light quarks have been compensated. The resulting isometric QCD values with QED corrections are

$$\begin{aligned} f_\pi^{\text{isoQCD}} &= 130.56(13)\,\text{MeV} \\ f_K^{\text{isoQCD}} &= 157.2(5)\,\text{MeV}. \end{aligned} \tag{5.34}$$

Using the experimental values in conjunction with the extrapolated lattice measurements we can extract the scale t_0 at the physical point.

$$\sqrt{t_0^{\text{phys}}} = \frac{F_\chi^{\text{cont}}(\phi_2^{\text{phys}}, \cdots)}{f_{\pi K}^{\text{isoQCD}}} \tag{5.35}$$

The physical point is defined by the meson masses, in this case the Pion and Kaon masses. Here we are using the masses reported in [86].

$$\begin{aligned} m_\pi^{\text{phys}} &= 134.9768(5)\,\text{MeV} \\ m_K^{\text{phys}} &= 497.611(13)\,\text{MeV} \end{aligned} \tag{5.36}$$

type	cut	χ^2/dof.	$\sqrt{t_0}f_\pi$	$\sqrt{t_0}f_{\pi K}$
Taylor	-	2.05		0.1083(3)
Taylor	$\beta > 3.4$	2.07		0.1088(4)
Taylor	$\beta > 3.5$	2.68		0.1084(5)
Taylor	$\phi_2 < 0.6$	1.77		0.1086(3)
Taylor	$\phi_2 < 0.4$	2.16		0.1086(4)
Taylor	$\beta > 3.4, \phi_2 < 0.6$	2.03		0.1090(5)
Taylor(4)	-	1.98		0.1081(3)
Taylor(4)	$\beta > 3.4$	1.69		0.1083(4)
Taylor(4)	$\beta > 3.5$	2.26		0.1078(5)
Taylor(4)	$\phi_2 < 0.6$	1.64		0.1084(4)
Taylor(4)	$\phi_2 < 0.4$	2.03		0.1083(4)
Taylor(4)	$\beta > 3.4, \phi_2 < 0.6$	1.43		0.1086(5)
SU(3) χPT	-	1.84		0.1081(3)
SU(3) χPT	$\beta > 3.4$	1.63		0.1085(4)
SU(3) χPT	$\beta > 3.5$	2.09		0.1081(5)
SU(3) χPT	$\phi_2 < 0.6$	1.50		0.1084(3)
SU(3) χPT	$\phi_2 < 0.4$	1.86		0.1084(4)
SU(3) χPT	$\beta > 3.4, \phi_2 < 0.6$	1.48		0.1088(5)
SU(3) χPT + $a^2m_\pi^2$	-	1.82		0.1085(4)
SU(3) χPT + $a^2m_\pi^2$	$\beta > 3.4$	1.77		0.1084(6)
SU(3) χPT + $a^2m_\pi^2$	$\beta > 3.5$	2.39		0.1080(8)
SU(3) χPT + $a^2m_\pi^2$	$\phi_2 < 0.6$	1.63		0.1085(5)
SU(3) χPT + $a^2m_\pi^2$	$\phi_2 < 0.4$	2.16		0.1086(7)
SU(3) χPT + $a^2m_\pi^2$	$\beta > 3.4, \phi_2 < 0.6$	1.18		0.1078(7)
SU(2) χPT	-	1.82	0.0933(4)	0.1074(3)
SU(2) χPT	$\beta > 3.4$	1.58	0.0937(4)	0.1079(4)
SU(2) χPT	$\beta > 3.5$	1.94	0.0933(6)	0.1075(5)
SU(2) χPT	$\phi_2 < 0.6$	1.26	0.0941(4)	0.1078(3)
SU(2) χPT	$\phi_2 < 0.4$	1.30	0.0945(5)	0.1079(4)
SU(2) χPT	$\beta > 3.4, \phi_2 < 0.6$	0.96	0.0947(6)	0.1085(5)

Table 5.2: Comparison of different chiral extrapolations at the reference point $(\phi_2, \phi_4) = (0.075, 1.12)$. A series of cuts are applied to the data before fitting. The quality of the fit is indicated by the value of the residue χ^2/dof.. The value for $\sqrt{t_0}f_{\pi K}$ ($\sqrt{t_0}f_\pi$) at the reference point is given.

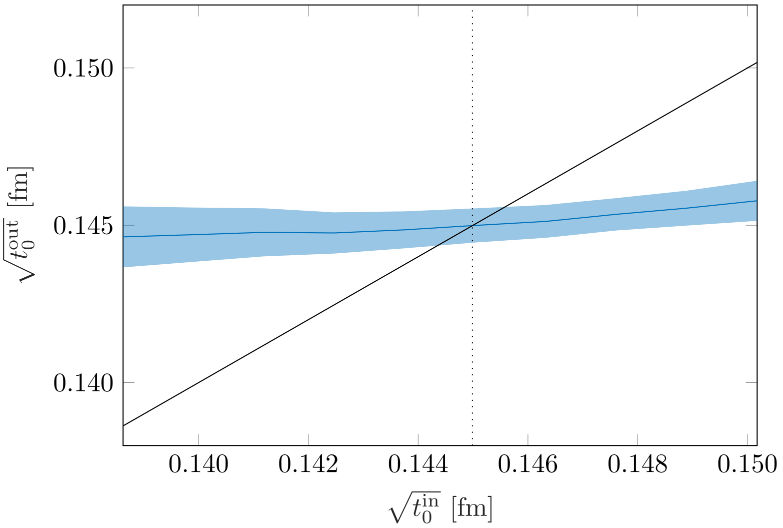


Figure 5.7: Dependence of the final scale $\sqrt{t_0^{\text{out}}}$ on the parameter used to define the physical point $(\phi_2^{\text{phys}}, \phi_4^{\text{phys}})$. The scale t_0^{out} is determined from the evaluation of the extrapolations in the continuum and at the physical point. The parameter t_0^{in} is used to define the physical point in the space of (ϕ_2, ϕ_4). The fixed point where $t_0^{\text{out}} = t_0^{\text{in}}$ is indicated by a vertical line.

In section 5.1 we transformed these masses into the dimensionless observables ϕ_2 and ϕ_4 using the flow scale t_0. The value of the physical point in the (ϕ_2, ϕ_4) space now depends on the flow scale t_0 that we want to extract in order to set the scale. The physical value for the flow scale t_0 is not known a priori and depends on the initial choice used in the determination of $(\phi_2^{\text{phys}}, \phi_4^{\text{phys}})$. The dependence of the scale t_0^{out} resulting from the fit on the scale t_0^{in} used in the definition of the physical point is shown in fig. 5.7. The fixed point where $t_0^{\text{in}} = t_0^{\text{out}}$ is indicated by the dotted line and marks the physical point. After setting initial values for the flow scale t_0^0 the fixed point is found using the following procedure.

1. Calculate the physical point in $(\phi_2^{\text{phys}}, \phi_4^{\text{phys}})$ space using the current approximation of the flow scale t_0^i
$$(\phi_2^{\text{phys}}, \phi_4^{\text{phys}}) = 8t_0^i \left(m_\pi^2, m_K^2 + \frac{1}{2} m_\pi^2 \right). \tag{5.37}$$
2. Shift the measurements to ϕ_4^{phys} as described in section 5.2.
3. Apply cuts to the data points.
4. Fit the shifted measurements for the scale setting quantity $\sqrt{t_0} f$ with the extrapolation functions $F_\chi(\phi_2, \cdots)$ presented in section 5.3.
5. Evaluate the fit function in the continuum and at ϕ_2^{phys}.
6. Compare $F_\chi^{\text{cont}}(\phi_2^{\text{phys}}, \cdots)$ to measurements of the decay constants using eq. (5.35) to get the scale t_0^{i+1}.
7. Repeat until convergence.

The fixed point in t_0 to a relative precision of $\left| \sqrt{t_0^i} - \sqrt{t_0^{i-1}} \right| < 10^{-4} \sigma \left(\sqrt{t_0^{i-1}} \right)$ is usually found in roughly 10 iterations. The correlated error analysis presented in section 3.6 is applied at every step of this process to obtain the final statistical error listed in table 5.3.

5.5 Determination of the Scale and its Uncertainties

Systematic uncertainties arise from the approximation of the chiral and continuum extrapolations. To estimate the magnitude of the systematic uncertainty we compare the different extrapolation techniques discussed in section 5.3. We use the quadratic and 4th Taylor expansion (Taylor, Taylor(4)), SU(3) chiral perturbation theory extrapolation (χPT) with an optional additional continuum term and the SU(2) chiral perturbation theory function (SU(2) χPT). All of these extrapolation functions are subjected to the fixed point iteration described in section 5.4 to determine the scale t_0. The results of the fixed point iteration are shown in table 5.3. Different cuts are applied to the data points before the chiral and continuum extrapolations are fitted. We successively remove the coarsest lattices and those furthest away from the physical point from the analysis. The cut $\phi_2 < 0.6$ (0.4) corresponds to restricting the Pion mass $m_\pi \lesssim 400\,\mathrm{MeV}$ (300 MeV).

In table 5.3 we present the physical point in (ϕ_2, ϕ_4)-space as well as the flow scale t_0 at that physical point. We also list characteristics of the shift and fits at the fixed point. The column labeled $\frac{\Sigma(\vec{n})}{\Sigma(\vec{0})}$ indicates the increase in the error due to the shift to the physical ϕ_4. This procedure is described in section 5.2. The statistical error is increased by 5% to 25% by the shifts. The fit is characterized by the χ^2/dof. as well as by the 'goodness of fit' (gof). The goodness of fit is the probability of measuring a χ^2 value that is greater than the one reported in the previous column. A probability close to 0.5 indicates that the measured χ^2 is in the center of the distribution.

We can see that on average the quality of the fit increases as we apply the cuts. The most notable improvement in the quality of the fit is obtained from the omission of the coarsest lattice and the symmetric ensembles. These are the ensembles the farthest away from their respective chiral and continuum extrapolations. One has to keep in mind that as more ensembles are removed from the analysis, the statistical error as well as the quality of the fit suffer. This can be seen when comparing the two cuts in ϕ_2. For all extrapolation techniques by far the best result is obtained by combining the cuts $\beta < 3.4$ and $\phi_2 < 0.6$.

The systematic error is estimated from the fluctuation of different extrapolations. We are only considering fits where the goodness of the fit is greater than 0.1. They are indicated by a star ($\star$) in table 5.3. The remaining fits fluctuate from $\sqrt{t_0^{\mathrm{min}}} = 0.1435\,\mathrm{fm}$ to $\sqrt{t_0^{\mathrm{max}}} = 0.1450\,\mathrm{fm}$ resulting in a systematic uncertainty of

$$\sigma_{\mathrm{syst}}\left(\sqrt{t_0^{\mathrm{phys}}}\right) = 0.0008\,\mathrm{fm} \tag{5.38}$$

The final value for the flow scale as well as its statistical error are taken from the two best fits. The SU(2) and SU(2) χPT fits with the strictest cut $\beta > 3.4$ & $\phi_2 < 0.6$ show a $\chi^2/\mathrm{dof} < 1$ and a goodness close to 0.5. The statistical uncertainty of these measurements is

$$\sigma_{\mathrm{stat}}\left(\sqrt{t_0^{\mathrm{phys}}}\right) = 0.0010\,\mathrm{fm}. \tag{5.39}$$

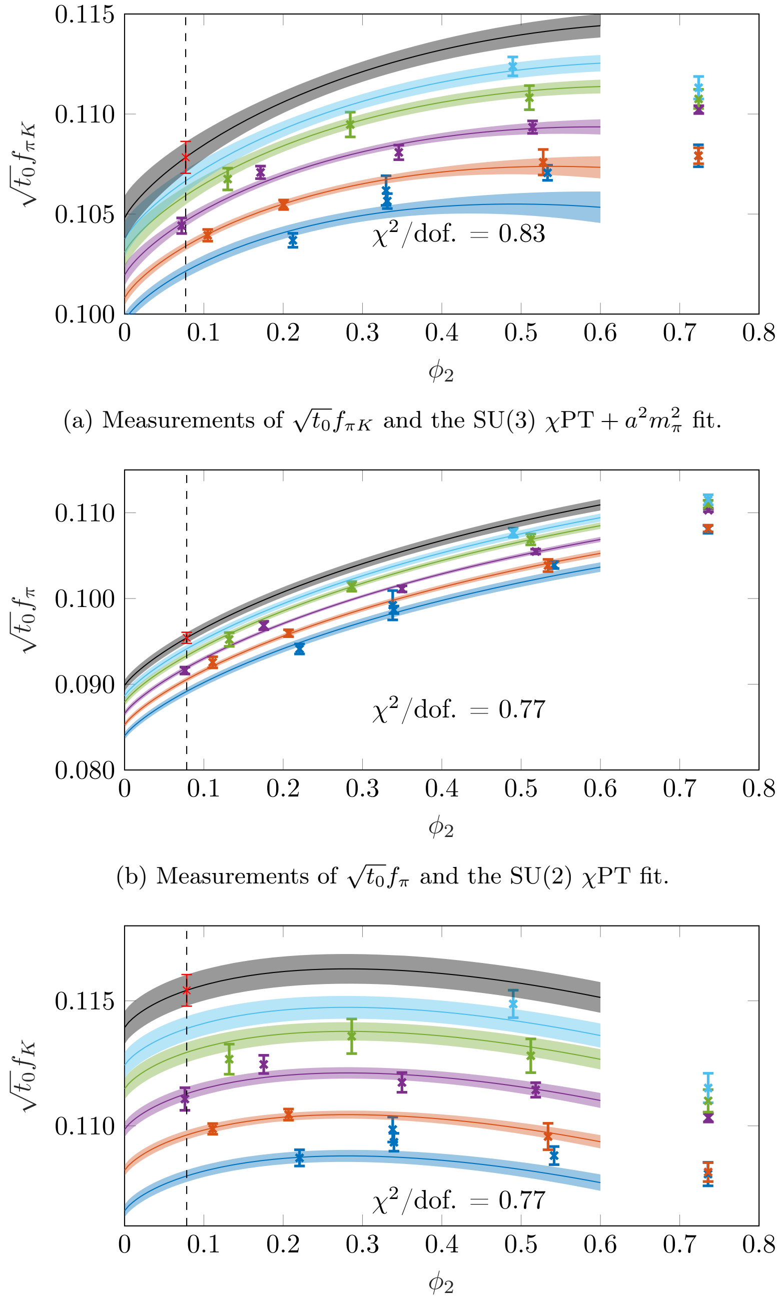
A weighted average of the best fits indicated by a diamond ($\diamondsuit$) in table 5.3 yields the scale

$$\sqrt{t_0^{\mathrm{phys}}} = 0.1441(10)(8)\,\mathrm{fm}. \tag{5.40}$$

The fits included in this determination of the intermediate scale are shown in fig. 5.8.

fit	cuts	$\frac{\Sigma(\vec{n})}{\Sigma(\vec{0})}$	$\frac{\chi^2}{\text{dof}}$	gof	ϕ_2^{phys}	ϕ_4^{phys}	$\sqrt{t_0^{\text{phys}}}$ [fm]	
Taylor (eq. (5.15))	-	1.11	1.79	0.02	0.0777(5)	1.095(8)	0.1441(6)	
	$\beta > 3.4$	1.12	1.84	0.04	0.0784(7)	1.105(10)	0.1447(7)	
	$\beta > 3.5$	1.25	1.99	0.04	0.0775(11)	1.092(15)	0.1439(10)	
	$\phi_2 < 0.6$	1.05	1.55	0.10	0.0780(6)	1.099(9)	0.1443(6)	
	$\phi_2 < 0.4$	1.06	1.81	0.08	0.0780(7)	1.099(10)	0.1444(7)	
	$\beta > 3.4$ & $\phi_2 < 0.6$	1.08	1.64	0.11	0.0787(8)	1.108(12)	0.1450(8)	⋆
Taylor(4) (eq. (5.16))	-	1.11	1.75	0.03	0.0775(5)	1.091(8)	0.1438(6)	
	$\beta > 3.4$	1.12	1.70	0.07	0.0779(7)	1.097(10)	0.1442(7)	
	$\beta > 3.5$	1.23	2.09	0.04	0.0768(10)	1.083(15)	0.1433(10)	
	$\phi_2 < 0.6$	1.05	1.48	0.13	0.0777(6)	1.095(9)	0.1441(6)	⋆
	$\phi_2 < 0.4$	1.06	1.87	0.08	0.0778(7)	1.096(10)	0.1442(7)	
	$\beta > 3.4$ & $\phi_2 < 0.6$	1.08	1.20	0.30	0.0783(8)	1.103(12)	0.1446(8)	⋆
χPT (eq. (5.22))	-	1.11	1.62	0.05	0.0774(5)	1.091(8)	0.1438(6)	
	$\beta > 3.4$	1.13	1.53	0.10	0.0780(7)	1.099(10)	0.1443(7)	
	$\beta > 3.5$	1.25	1.68	0.10	0.0769(11)	1.083(15)	0.1433(10)	
	$\phi_2 < 0.6$	1.05	1.33	0.19	0.0777(6)	1.095(8)	0.1441(6)	⋆
	$\phi_2 < 0.4$	1.06	1.61	0.13	0.0778(7)	1.097(10)	0.1442(7)	⋆
	$\beta > 3.4$ & $\phi_2 < 0.6$	1.08	1.19	0.30	0.0784(8)	1.105(12)	0.1447(8)	⋆
$\chi\text{PT} + a^2m_\pi^2$ (eq. (5.31))	-	1.10	1.60	0.06	0.0780(7)	1.099(10)	0.1444(7)	
	$\beta > 3.4$	1.14	1.57	0.10	0.0779(9)	1.098(14)	0.1443(9)	
	$\beta > 3.5$	1.20	2.04	0.05	0.0779(12)	1.098(18)	0.1442(12)	
	$\phi_2 < 0.6$	1.07	1.40	0.17	0.0781(9)	1.101(13)	0.1445(9)	⋆
	$\phi_2 < 0.4$	1.10	1.73	0.11	0.0785(12)	1.107(17)	0.1449(11)	⋆
	$\beta > 3.4$ & $\phi_2 < 0.6$	1.12	0.83	0.56	0.0770(11)	1.086(16)	0.1435(11)	⋆ ◇
SU(2) χPT (eq. (5.28))	-	1.15	1.63	0.01	0.0768(6)	1.082(8)	0.1432(6)	
	$\beta > 3.4$	1.18	1.66	0.02	0.0775(7)	1.091(11)	0.1438(7)	
	$\beta > 3.5$	1.40	1.61	0.06	0.0760(12)	1.071(18)	0.1425(12)	
	$\phi_2 < 0.6$	1.09	1.09	0.34	0.0772(6)	1.088(9)	0.1436(6)	⋆
	$\phi_2 < 0.4$	1.09	1.11	0.34	0.0774(7)	1.091(11)	0.1438(7)	⋆
	$\beta > 3.4$ & $\phi_2 < 0.6$	1.13	0.77	0.72	0.0783(8)	1.104(12)	0.1447(8)	⋆ ◇

Table 5.3: Overview of different chiral and continuum models discussed in section 5.3. We also present different cuts to the data. The results shown here are located at the physical point defined using the fixed point procedure described in section 5.4. Each method results in a slightly different physical point defined by $(\phi_2^{\text{phys}}, \phi_4^{\text{phys}})$. The relative amplification of the statistical error as a result of the shift to the physical point is listed in the third column. To check the quality of the fit we report the χ^2/dof. as well as the goodness of the fit. Finally we give the flow scale $\sqrt{t_0}$ at the physical point for each of the extrapolations. Measurements indicated with a star ($\star$) are included in the estimation of the systematic error. Those indicated with a diamond ($\diamondsuit$) make up the average and statistical error.



(a) Measurements of $\sqrt{t_0}f_{\pi K}$ and the SU(3) χPT $+\, a^2 m_\pi^2$ fit.

(b) Measurements of $\sqrt{t_0}f_{\pi}$ and the SU(2) χPT fit.

(c) Measurements of $\sqrt{t_0}f_{K}$ and the SU(2) χPT fit.

Figure 5.8: Combined chiral and continuum extrapolations at the physical point for the measurements that constitute the result given in eq. (5.40). We apply the strictest cuts to the data, i.e. $\beta > 3.4$ & $\phi_2 > 0.6$. Figure (a) shows measurements for $\sqrt{t_0}f_{\pi K}$ as well as the fit labeled SU(3) χPT $+\, a^2 m_\pi^2$ defined by eqs. (5.22) and (5.31). In figures (b) and (c) we present measurements of $\sqrt{t_0}f_{\pi}$ and $\sqrt{t_0}f_{K}$ respectively. We also show the SU(2) χPT fits given by eqs. (5.28) and (5.29). The quality of the fit is indicated by the χ^2 values.

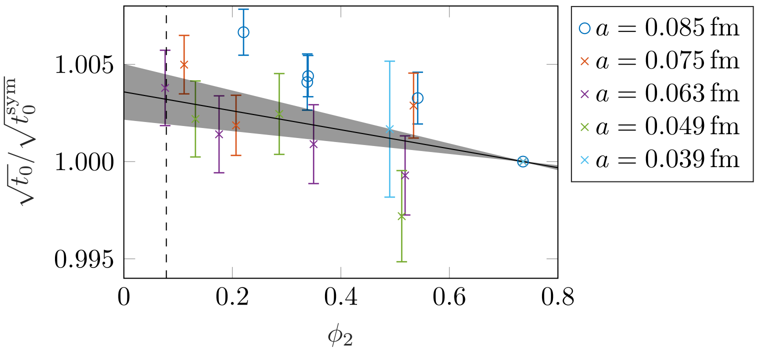


Figure 5.9: Flow scale $\sqrt{t_0}$ as a function of ϕ_2. The measurements are normalized by the values at the symmetric point to eliminate effects of the lattice spacing a and allow for a combined fit. A fit with eq. (5.41) is shown in the background. Measurements indicated with circles are not included in the fit.

5.6 Lattice Spacing

In the previous section we have calculated the intermediate flow scale t_0 at the physical point. We can now use this scale to extract the lattice spacing a in physical units. To do that we extract measurements of $\sqrt{\frac{t_0}{a^2}}$ to the physical point. In order to extrapolate all measurements at the same time, we normalize by the values at the symmetric point for each lattice spacing. These normalized values are shown in fig. 5.9. These normalized measurements are then modeled with

$$F(\phi_2, P) = \sqrt{1 + P\,(\phi_2 - \phi_2^{\mathrm{sym}})}. \tag{5.41}$$

The parameter P is determined from a fit to the data. The measurements of the coarsest lattice spacing ($a = 0.085\,\mathrm{fm}$) lie systematically above the others. The other values agree within their statistical fluctuation. For that reason the measurements on the coarsest lattice spacing are omitted from the fit.

Using the fit function evaluated at the physical point ϕ_2^{phys} and removing the normalization by multiplying the symmetric measurements we obtain the value for $\sqrt{\frac{t_0}{a^2}}$ at the physical point

$$\left(\sqrt{\frac{t_0}{a^2}}\right)^{\mathrm{phys}} = \sqrt{\frac{t_0^{\mathrm{sym}}}{a^2}} \cdot F(\phi_2^{\mathrm{phys}}, P). \tag{5.42}$$

We now have the value of $\sqrt{\frac{t_0}{a^2}}$ at the physical point for each of the inverse couplings β. Together with the determination of the scale t_0^{phys} in eq. (5.40) we can extract the lattice spacing

$$a = \frac{\sqrt{t_0^{\mathrm{phys}}}}{\left(\sqrt{\frac{t_0}{a^2}}\right)^{\mathrm{phys}}}. \tag{5.43}$$

The lattice spacing for each of the inverse couplings is given in table 5.4. The errors consist of the statistical, correlated error analysis laid out in section 3.6 as well as the systematic error calculated in section 5.5.

β	$\frac{t_0^{\mathrm{sym}}}{a^2}$	a [fm]
3.40	2.876(10)	0.0848(6)(4)
3.46	3.688(13)	0.0748(5)(4)
3.55	5.16(2)	0.0632(4)(3)
3.70	8.59(3)	0.0490(3)(3)
3.85	13.96(7)	0.0384(3)(2)

Table 5.4: The lattice spacing a is given in physical units for each inverse coupling β considered in this analysis. The lattice spacing is calculated from the intermediate scale given in eq. (5.40), the fit shown in fig. 5.9 and measurements of $\frac{t_0}{a^2}$ at the symmetric point. These measurements are given in the second column. The lattice spacing is given in the last column. Its statistical and systematic uncertainties are given in parenthesis and originate from the uncertainties of the scale t_0 given in eq. (5.40).

6 Conclusion

In the first part we computed the scale for simulations of QCD with $2+1$ flavors of NP improved fermions as used by the CLS group [108]. Compared to an earlier determination [1] we are using

- ensembles with five different lattice spacings including one that is smaller than the lattice spacings previously used,
- ensembles that are closer to the physical point,
- a combination of measurements from different groups [59, 61–63] to maximize statistics using measurements that are already available,
- measurements of the quark mass derivatives on most ensembles and subsequent modeling of the derivatives for all values of ϕ_2,
- a number of models for the chiral and continuum extrapolations to estimate and constrain the systematic uncertainty.

Compared to the 2017 analysis [1], this allowed us to significantly improve the control of the systematic uncertainties for the chiral and continuum extrapolations. The inclusion of finer lattices and ensembles close to the physical point is effective at reducing the uncertainty. Ensembles like `J500` and `J501` reduce the distance the continuum extrapolation has to cover. Likewise, the ensembles `E250`, `E300` and `D452` close to the physical point help to stabilize the chiral extrapolation. Using these ensembles allows us to apply cuts to the measurements. In table 5.3 we can see that removing ensembles far away from the physical point and the continuum does not lead to a declining quality of the fit. Additionally, using a wider range of different extrapolation techniques we are more confident in the correct estimation of the systematic uncertainty.

Another difference between the determination in 2017 and this analysis is the calculation of the quark mass derivatives. In the previous analysis [1] the quark mass derivatives are calculated using two ensembles with varying quark masses. These measurements are then extrapolated to the other ensembles using chiral perturbation theory. Here, we use measurements of the derivatives on the majority (19/20) of the ensembles and a model to fit the derivatives. This combined description of the derivatives results in an average relative error of about 35% compared to 50% in [1]. Measurements of the derivatives on ensembles close to the physical point, where the simulation is computationally expensive, often have larger statistical uncertainties than the measurements at the symmetric point. These measurements, in particular, benefit from the combined description, as the uncertainty of the fit is smaller than the uncertainty of these individual measurements (see fig. 5.4). Additionally, we are able to extrapolate the quark mass derivatives to ensembles where they have not been measured.

As a result of these improvements, the statistical uncertainty given in eq. (5.40) is reduced by about 29% compared to the previous determination in [1]. However, using a larger set of

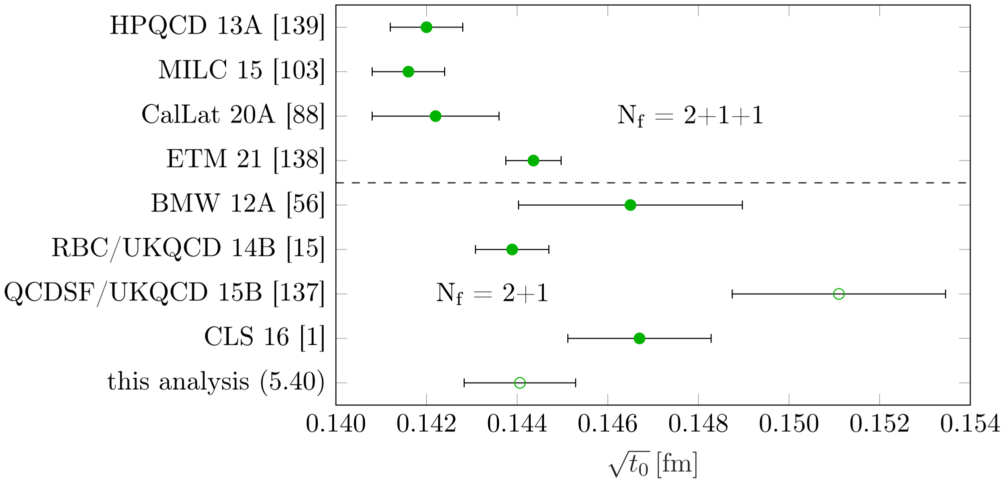


Figure 6.1: Comparison of the intermediate flow scale calculated by different groups. The values are taken from [9] and the references therein. Empty symbols indicate unpublished results.

chiral and continuum extrapolations the estimation of the systematic uncertainty is increased by 13%. The resulting relative precision is 0.89%, where the statistical and systematic uncertainty contribute roughly equal parts.

To increase the precision even further one would need to simulate ensembles with a fine lattice spacing close to the physical point. In particular, an extension of the $\beta = 3.85$ trajectory towards the physical point would help to stabilize the extrapolations. Advances in the simulation algorithms would improve the feasibility of simulations with smaller lattice spacings. See part II for our efforts in this regard. Apart from additional simulations, there are theoretical improvements. A more precise understanding of the behavior along the continuum and chiral extrapolations results in a more precise determination of the scale at the physical point. One could for example include higher terms such as the ones determined in [132, 133] in the continuum extrapolation. Currently, the analysis reaches around 1% accuracy. Going further, QED effects become relevant and lattice simulations including QED [32, 134–136] are needed.

To further increase the precision, one could consider different phenomenological scales to determine the intermediate scale t_0. If, for instance, one can measure $\sqrt{t_0}m_p$ precisely on the lattice, one can benefit from the exceptional precision of the experimental determination of the proton mass.

In fig. 6.1 we show how our result compares to previous determinations of the scale. The updated value is in agreement with other (published) results. In particular, the most precise determinations by the ETM collaboration [138] and the RBC/UKQCD collaboration [15] are matched very closely.

Part II

FETI Algorithm

7 | Introduction

In lattice QCD simulations the inversion of the Dirac operator is the most computationally expensive part. Lattice QCD strives towards larger lattice sizes and lighter quark masses. Recent developments have in particular called for very large lattice sizes [140, 141]. The CLS group also generated ensembles at physical Pion masses (see fig. 3.1 and [107, 142]). Both of these requirements increase the difficulty of the solution of the Dirac equation. Increasing the lattice size directly increases the size of the system. Lowering the bare quark mass decreases its condition number. For these reasons, the inversion of the Fermion matrix given by the Dirac operator in eq. (1.16) is a bottleneck of the simulation. Algorithmic advancements in the solution of the Dirac equation greatly influence the quality and feasibility of lattice QCD simulations.

Because of the size of the Dirac operator, explicit matrix inversion techniques are unrealistic. While the Dirac operator itself is sparsely populated, its inverse is not. Hence, explicitly calculating and saving the inverse Dirac operator exceeds the memory available today. For this reason we resort to iterative techniques that will be introduced in chapter 8. These techniques approximate the application of the inverse Dirac operator each time it is needed. Each time the Dirac equation must be solved, the iteration is started from the beginning. This process is very computationally intensive even on the most advanced parallel computers. Algorithmic improvements go a long way in increasing the statistics and feasibility of lattice QCD simulations.

One way to tackle such complicated matrix inversion problems are domain decomposition methods. Here the problem is divided into several sub-problems and distributed among a number of computation nodes. This limits the workload of each individual node and can, if done efficiently, drastically speed up the computation. An overview of domain decomposition methods is given in [65]. When using domain decomposition methods one has to consider the overhead introduced by these methods and their scaling behavior. Another key distinction between different domain decomposition solvers is the condition number of the decomposed system. If the decomposed system is very ill-conditioned, it can be more difficult to solve than the original system. The communication between different computation nodes also plays an important role when analyzing these methods. It is in these key points where different domain decomposition algorithms vary.

Two such algorithms will be introduced in chapter 9. The first one, which we will consider as a reference point, is the Schwarz Alternating Procedure. It is implemented as part of the `openQCD` package listed in table A.1 and has been successfully employed in lattice QCD applications [143, 144]. The second one is called **F**inite **E**lement **T**ear and **I**nterconnect (FETI) algorithm [65, 145] and is the main focus of part II of this work. We implemented the algorithm on top of the `openQCD` package [123]. The FETI algorithm has been used successfully in engineering applications such as linear elasticity [66, 67].

In chapter 10 we will have an detailed look at the FETI algorithm in a lattice QCD application. We will analyze its convergence as well as the behavior of the subsystems. Several preconditioning techniques and the comparison to current state-of-the-art methods are considered.

8 | Linear Algebra Solvers

As we have seen in the previous section, it is of great importance to be able to efficiently solve linear systems of the form

$$A\hat{x} = f. \tag{8.1}$$

In linear algebra there exist numerous methods to calculate the inverse of an operator A. An overview of several such methods is given in [64, 146, 147].

The matrices we are interested in here are prohibitively large. The dimension can reach up to 10^9 at which point it would take millions of Terabytes to fully store the operator matrix. Methods that make use of the explicit form of the matrix are ruled out in this case. This includes all factorization (QR, SVD, LU, etc.) and elimination (Gauss, etc.) methods that are very common in linear algebra.

The operator used in this application is the Dirac operator. The definition is given in section 1.2 eq. (1.16). It is sparsely populated. It has entries only on the diagonal and for the two neighbors in each direction. This means that it consists of only $\mathcal{O}\left(m\right)$ entries as opposed to $\mathcal{O}\left(m^2\right)$, if m is the dimension of the matrix.

Given the huge dimension of the operator and its sparse population we can only efficiently implement the application of the operator on a vector. For this reason we need to find a solver that only utilizes the application of the operator rather than its explicit structure. These solvers are called Krylov subspace solvers and are a subclass of the iterative solvers.

An introduction to selected iterative solvers and preconditioning techniques used here is given in sections 8.1 and 8.2.

Krylov solvers are iterative solvers that repeat some process

$$x_{k+1} = P\left(A, f, x_k, x_{k-1}, \cdots\right) \tag{8.2}$$

to update the current approximation to the exact solution

$$\hat{x} = A^{-1} f. \tag{8.3}$$

This update process is constructed such that upon repeated application the solution converges to the exact solution.

$$x_k \xrightarrow{k\to\infty} \hat{x} \tag{8.4}$$

There exist a variety of different iterative solvers [147], each characterized by its update process P that determines the step $x_k \to x_{k+1}$. To measure the accuracy of an approximate solution x_k we use the residue

$$\rho = f - A x_k. \tag{8.5}$$

If the residue gets small, in the sense that

$$\|\rho\| < \epsilon \|f\| \tag{8.6}$$

for some small ϵ, the difference of the approximate solution x_k to the exact solution is bounded by

$$\|x_k - \hat{x}\| < \epsilon\kappa(D) \|\hat{x}\| , \tag{8.7}$$

with the condition number

$$\kappa(D) = \|D\| \left\|D^{-1}\right\| . \tag{8.8}$$

For the types of solvers used here (Krylov solvers) the condition number measures how easily the matrix can be inverted and how fast the iterative algorithms converge. For the unit matrix, which is its own inverse, the condition number is one, while for a singular matrix the condition number is infinite. The matrices discussed here lie somewhere in between and can be compared using their condition number. Preconditioners are operations that transform the original linear system in order to decrease its condition number. The preconditioned system can then be solved and the original solution recovered. Preconditioning techniques are very important when dealing with matrices that are close to singular, i.e. matrices with a large condition number.

8.1 Krylov Subspace Solver

Krylov solvers are suited for large, sparse linear systems, because only the application of the operator is required. They approximate the solution $\hat{x} = A^{-1}f$ by some polynomial $\hat{x} \approx p\,(A)\,f$. The approximate solution x_k is an element of the Krylov space

$$\mathcal{K}_k = \left\{f, Af, A^2 f, \cdots, A^{k-1} f\right\} \tag{8.9}$$

spanned by repeated application of the operator on the vector f. Iterative Krylov solvers such as the ones described in sections 8.1.1 to 8.1.3 construct an approximate solution vector in this space. They differ in the construction procedure, convergence properties and computational requirements. The algorithms listed below are described in [64, 147, 148]. Further improvements such as restarting and single precision acceleration are discussed in [64].

8.1.1 Steepest Descend and Conjugate Gradient

The steepest descend and conjugate gradient (CG) algorithms [147–149] work for hermitian matrices $A = A^\dagger$. If the system in question is not symmetric, we solve the system $A^\dagger Ax = A^\dagger f$. Therefore, without loss of generality we assume that A is symmetric here. The central idea behind these algorithms is minimizing the quantity

$$\phi(x) = \frac{1}{2}\,(x, Ax) - (x, f) \tag{8.10}$$

with respect to x. Here (a, b) is the scalar product of the vectors a and b. Minimizing $\phi(x)$ is equivalent to solving the system $Ax = f$ since $x_{\min} = A^{-1}f$. At some point x_k we calculate the steepest descent and use that as a direction for the next iteration. The gradient is

$$-\nabla\phi(x_k) = f - Ax_k = \rho_k, \tag{8.11}$$

which is exactly the residue ρ_k of the current solution. There is then a point $x_k + \alpha\rho_k$ where $\phi\,(x_k + \alpha\rho_k) < \phi\,(x_k)$ We can now minimize

$$\phi\,(x_k + \alpha\rho_k) = \phi\,(x_k) - \alpha\,(\rho_k, \rho_k) + \frac{1}{2}\alpha^2\,(\rho_k, A\rho_k) \tag{8.12}$$

with respect to α to get the optimal distance

$$\alpha = \frac{(\rho_k, \rho_k)}{(\rho_k, A\rho_k)} \tag{8.13}$$

to shift the current solution

$$x_{k+1} = x_k + \alpha\rho_k. \tag{8.14}$$

The steepest descend algorithm consists of the following steps that are repeated until convergence.

1. Calculate the residue ρ_k.
2. Calculate the shift distance α according to eq. (8.13).
3. Shift the current solution in the ρ_k direction using eq. (8.14).

The optimal value for $\phi(x)$ is $\phi^{\text{opt}}(x) = -\frac{1}{2}\left(f, A^{-1}f\right)$. The distance from this optimal value is reduced by the CG iteration according to

$$\left(\phi(x_k) + \frac{1}{2}\left(f, A^{-1}f\right)\right) \leq \left(1 - \frac{1}{\kappa(A)}\right)\left(\phi(x_{k-1}) + \frac{1}{2}\left(f, A^{-1}f\right)\right). \tag{8.15}$$

In that way the convergence is guaranteed if $\kappa(A) > 0$. The iterative improvement is mediated by the condition number $\kappa(A)$. The convergence is bound by

$$\frac{\|\rho_k\|}{\|\rho_0\|} \leq \left(1 - \frac{1}{\kappa(A)}\right)^k. \tag{8.16}$$

Note that the optimum in eq. (8.12) is found in the direction of ρ_k. Choosing a different search direction for this optimization can improve convergence and leads to the conjugate gradient (CG) algorithm [147, 149]. Here, $\phi(x)$ is minimized along some search direction p_k resulting in the following conjugate gradient procedure.

Starting with the search direction $p_0 = \rho_0$

1. Calculate the shift distance
$$\alpha = \frac{(\rho_k, \rho_k)}{(p_k, Ap_k)}. \tag{8.17}$$
2. Update the current solution and the residue
$$\begin{aligned} x_{k+1} &= x_k + \alpha p_k \\ \rho_{k+1} &= \rho_k - \alpha A p_k. \end{aligned} \tag{8.18}$$
3. Calculate the new search direction using
$$p_{k+1} = \rho_{k+1} + \frac{(\rho_{k+1}, \rho_{k+1})}{(\rho_k, \rho_k)} p_k. \tag{8.19}$$

The convergence of the CG algorithm is given by [148]

$$\frac{\|\rho_k\|}{\|\rho_0\|} \leq \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1}\right)^k \tag{8.20}$$

again using the condition number $\kappa(A)$.

8.1.2 Minimal Residual

The Minimal Residual (MRES) algorithm [150] is very similar to the CG algorithm. However, it can deal with non-hermitian matrices. The cost function $\phi(x_k)$ that is minimized is set to the norm of the residual

$$\phi(x_k) = \|f - Ax_k\|^2 = \|\rho_k\|^2 . \tag{8.21}$$

Minimizing

$$\phi\left(x_k + \alpha\rho_k\right) = \phi\left(x_k\right) - 2\alpha\left(\rho_k, A\rho_k\right) + \alpha^2\left(A\rho_k, A\rho_k\right) \tag{8.22}$$

with respect to α we get

$$\alpha = \frac{(\rho_k, A\rho_k)}{(A\rho_k, A\rho_k)} \tag{8.23}$$

From this point on the algorithm works in exactly the same way as the CG algorithm presented in the previous section. If μ is the smallest eigenvalue of the hermitian matrix $\frac{1}{2}\left(A + A^\dagger\right)$ and $\sigma = \|A\|^2$ then the MRES algorithm converges as

$$\|\rho_{k+1}\| \leq \left(1 - \frac{\mu^2}{\sigma^2}\right)^{\frac{1}{2}} \|\rho_k\| . \tag{8.24}$$

8.1.3 Generalized Conjugate Residue

The generalized conjugate residue (GCR) algorithm [151–153] works for arbitrary matrices A. The requirement that A must be hermitian, that existed for the CG algorithm, is no longer needed. The presentation of the GCR algorithm is taken from [144].

The GCR generates approximate solutions $x_k \in \mathcal{K}_k, k = 1, 2, 3, \ldots$ in the Krylov space. We will call the basis of the Krylov space v_l. It will be useful to define the conjugate space $\mathcal{L}_k$.

$$\mathcal{K}_k = \{v_0, v_1, \cdots, v_{k-1}\} = \left\{f, Af, \cdots, A^{k-1}f\right\} \tag{8.25}$$

$$\mathcal{L}_k = \{\chi_0, \chi_1, \cdots, \chi_{k-1}\} = A\mathcal{K}_k \tag{8.26}$$

Since the solution is in the Krylov space $\mathcal{K}$, we can express is using the basis vectors v_l

$$x_k = \sum_{i=0}^{k-1} c_i v_i \tag{8.27}$$

For the associated residue ρ_k we get

$$\rho_k = f - Ax_k = f - \sum_{i=0}^{k-1} c_i A v_i . \tag{8.28}$$

The GCR algorithm minimizes the norm of the residue

$$\|\rho_k\|^2 = \|f\|^2 - 2\sum_{i=0}^{k-1} c_i \left(f, Av_i\right) + \sum_{i,j=0}^{k-1} c_i c_j \left(Av_i, Av_j\right) . \tag{8.29}$$

when shifting the current solution vector x_k in the direction v_{k-1}. Minimizing with respect to c_l we get

$$\begin{aligned} 0 = \frac{\partial \|\rho_k\|^2}{\partial c_l} &= -2\left(f, Av_l\right) + 2\sum_{i=0}^{k-1} c_i \left(Av_i, Av_l\right) \\ &= -2\left(f, Av_l\right) + 2\left(Ax_k, Av_l\right) . \end{aligned} \tag{8.30}$$

Using the definition of the basis vectors in eqs. (8.25) and (8.26), i.e. $\chi_l = Av_l$, we arrive at the condition

$$(Ax_k, \chi_l) = (f, \chi_l)\,. \tag{8.31}$$

that implicitly defines the vector x_k that minimizes the residue $\|\rho_k\|^2$ in the search direction v_{k-1}. The condition is solved for

$$Ax_k = \sum_{i=0}^{k-1} \underbrace{(f, \chi_i)}_{\alpha_i} \chi_i = \sum_{i=0}^{k-1} \alpha_i \chi_i. \tag{8.32}$$

Insert into eq. (8.28) can express the residue in terms of the right-hand side f and the basis vectors χ_l

$$\rho_k = f - \sum_{i=0}^{k-1} (f, \chi_i)\, \chi_i. \tag{8.33}$$

The residue is the projection of the right-hand side f onto the orthogonal complement of $\mathcal{L}_k$. The GCR algorithm calculates this residue before the approximate solution for this step x_k is known. We then use the residue ρ_k to calculate the next basis vector to the conjugate space $\mathcal{L}_k$.

$$\chi_i = \sum_{j=0}^{l} \beta_{ij} A\rho_j, \quad i = 0, 1, 2, \cdots \tag{8.34}$$

The coefficients β_{ij} are defined through the orthonormality conditions with the previous basis vectors.

$$(\chi_l, \chi_j) = \delta_{ij}. \tag{8.35}$$

Inserting eq. (8.34) into eq. (8.32) and applying A^{-1} to both sides leads to an explicit equation for the current approximate solution

$$x_n = \sum_{i=0}^{n-1} \sum_{j=0}^{l} \alpha_i \beta_{lj} \rho_j. \tag{8.36}$$

The GCR algorithm consists of the following steps.

1. Calculate the coefficients $\alpha_i = (f, \chi_i)\,, \quad i = 0, 1, \cdots, k-1$ from the basis vectors calculated in previous steps.

2. Calculate the residue ρ_k and the new approximation x_k.

3. Calculate the coefficients β_{ij} and the new search direction v_k using ρ_k, the previous basis vectors and eq. (8.34).

Because the GCR algorithm is used extensively in this application we also give it in algorithmic form in algorithm 1.

It is shown in [64] that the GCR algorithm converges with

$$\|\rho_k\|^2 \leq \kappa(V) \left(1 + \frac{m}{M}\right)^{-k} \|f\|^2 \tag{8.37}$$

where V is an invertible matrix according to the diagonalization $A = V\Lambda V^{-1}$ and M and m define a disc with radius m and centered at M that contains all eigenvalues of A. The convergence rate $\frac{m}{M} \sim \frac{2}{\kappa(A)}$ is linked to the condition number $\kappa(A)$ of the operator.

Algorithm 1 GCR

```
$\rho_0 = f - Ax_0, \quad v_0 = \rho_0$
for $i = 0, 1, \cdots$, until convergence do
    $\alpha_i = \frac{(\rho_i, Av_i)}{(Av_i, Av_i)}$
    $x_{i+1} = x_i + \alpha_i v_i$
    $\rho_{i+1} = \rho_i - \alpha_i A v_i$
    $v_{i+1} = \rho_{i+1}$
    for $j = 0, 1, \cdots, l$ do
        $\beta_{ij} = -\frac{(A\rho_{j+1}, Av_l)}{(Av_l, Av_l)}$
        $v_{i+1} = v_{i+1} + \beta_{ij} v_j$
    end for
end for
```

We can further improve the quality of the solver by periodically restarting the algorithm. Restarting means that after a fixed number of iterations n_{restart} we forget about the search directions $v_i, i < n_{\text{restart}}$ used so far and starting the algorithm from the current solution. In this way we avoid the accumulation of errors caused by limited floating point precision. This also decreases the memory requirements of the algorithm as fewer Krylov vectors v_i need to be saved.

The convergence properties of the different Krylov solvers depend on the characteristics of the linear system to be solved and the resulting optimization landscape. An analysis for the linear systems relevant for this application is presented in section 10.3.2. There we find that the GCR solver is superior to the others in terms of stability and convergence rate.

8.2 Preconditioning

Preconditioning is the explicit or implicit transformation of a system of equations designed to improve the convergence of iterative solvers. There exist a wide variety of such techniques, some of which are presented in [147]. Almost universally choosing a suitable preconditioner is far more important than choosing the right solver. In this section we will give a general, theoretical introduction using a generic preconditioning operator M and discuss how the iterative solvers introduced in section 8.1 can be extended to cover preconditioned systems. Several preconditioners for the system used in this application are discussed in sections 9.2.6, 10.2.2 and 10.3.4

One type of explicit preconditioner is to scale all rows and/or columns of A such that the diagonal entries are one or close to one. This transformation may lead to a faster convergence using a Krylov space solver. However, this is not guaranteed. A brief analysis of this preconditioner is given in section 10.3.4.

More complicated, implicit preconditioners are of the form

$$M_L^{-1} A M_R^{-1} u = M_L^{-1} f, \quad u = M_R x. \tag{8.38}$$

The left and right preconditioning operators M_L and M_R can be complicated transformations. They may involve iterative procedures, integral calculations and other forms of transformations. In this case no linear matrix M can be defined. Instead, the preconditioner is given by a non-linear function $M^{-1}(u)$. The notation will be abused in a way that $M^{-1}u$ refers to the application of the preconditioner M^{-1} either being a linear matrix or a non-linear function. By design any solution to the preconditioned system (8.38) is also a solution to the original system. However, if we chose suitable preconditioning operators, we can design the preconditioned system to be advantageous to the original system. In particular this means the condition number of the preconditioned system is smaller: $\kappa\left(M_L^{-1}AM_R^{-1}\right) \leq \kappa\left(A\right)$.

In the following we will focus on left preconditioners. This amounts to setting $M_R = \mathbb{1}$. The left preconditioner can be understood as selecting a search direction in which to minimize the residue. We are interested in preconditioners that approximate the inverse operator $M \approx A^{-1}$. A good preconditioner finds a balance between the cost of the application and the quality of the approximation of A^{-1}. A suitable measure of how close the preconditioner matches the inverse operator is the Frobenius norm of the residual matrix

$$F(M) = \|\mathbb{1} - AM\|_F^2 . \tag{8.39}$$

Since, in general, the preconditioning matrix M is dense, it is computationally intensive to compute this norm. For that reason we compare different preconditioners by the convergence behavior of the preconditioned iterative solvers. To that end we extend the solvers to work on preconditioned systems. In section 8.2.1 we focus on the preconditioned version of the GCR algorithm introduced in section 8.1.3. A comparison of different preconditioners in a QCD application is given in section 10.2.2.

8.2.1 Preconditioned GCR

In each step of the algorithm presented in section 8.1.3 we minimize the residue when shifting the current solution vector x_k in the search direction v_k calculated from the residue ρ_k. In the preconditioned version we now calculate the search direction $\bar{v}_k$ from the vector $M^{-1}\rho_k$. We get the preconditioned algorithm by making the following substitutions

$$\begin{aligned} A &\to AM^{-1} \\ x_k &\to Mx_k. \end{aligned}$$

We can then define the new search direction $\bar{v}_k = M^{-1}v_k$, which leads to the preconditioned GCR algorithm listed in algorithm 2.

Algorithm 2 Preconditioned GCR.

```
$\rho_0 = f - Ax_0, \quad \bar{v}_0 = M^{-1}\rho_0$
for $i = 0, 1, \cdots$, until convergence do
    $\alpha_i = \frac{(\rho_i, A\bar{v}_i)}{(A\bar{v}_i, A\bar{v}_i)}$
    $x_{i+1} = x_i + \alpha_i \bar{v}_i$
    $\rho_{i+1} = \rho_i - \alpha_i A\bar{v}_i$
    $\bar{v}_{i+1} = M^{-1}\rho_{i+1}$
    for $j = 0, 1, \cdots, l$ do
        $\beta_{ij} = -\frac{\left(AM^{-1}\rho_{j+1}, A\bar{v}_l\right)}{(A\bar{v}_l, A\bar{v}_l)}$
        $\bar{v}_{i+1} = \bar{v}_{i+1} + \beta_{ij}\bar{v}_j$
    end for
end for
```

8.3 Deflation

Deflation is a technique to drastically speed up the inversion of linear systems. It is used in lattice QCD applications and described in [64, 69, 70]. It works by separating a part of the solution, where the inverse can be explicitly calculated. The space in which the explicit solution resides is called the deflation subspace $\mathcal{S}$. If this space is chosen appropriately the condition number of the remaining system is smaller than the original. The deflation subspace is spanned by the orthonormal basis vectors ϕ_k

$$\mathcal{S} = \operatorname{span}\left(\phi_1, \phi_2, \cdots, \phi_N\right) \tag{8.40}$$

For this general overview of the technique we will not further specify the deflation subspace. Explicit examples of the construction of the deflation subspace are presented in sections 9.1.1 and 9.2.6.

We begin by defining the projector onto the deflation subspace

$$Px = \sum_{k=1}^{N} \phi_k \left(\phi_k, x\right). \tag{8.41}$$

The application of the linear operator A onto the basis vectors of the deflation subspace can be expressed by the matrix

$$E_{kl} = \left(\phi_k, A\phi_l\right). \tag{8.42}$$

This matrix is sometimes referred to as the “little operator”. It is assumed that E is invertible and explicitly known. For normalized vectors ϕ, its inverse can be calculated as follows

$$\left(E^{-1}\right)_{kl} = \left(\phi_k, A^{-1}\phi_l\right). \tag{8.43}$$

The linear operators

$$P_L x = x - \sum_{k,l=1}^{N} A\phi_k \left(E^{-1}\right)_{kl} \left(\phi_l, x\right) \tag{8.44}$$

$$P_R x = x - \sum_{k,l=1}^{N} \phi_k \left(E^{-1}\right)_{kl} \left(\phi_l, Ax\right) \tag{8.45}$$

$$\tag{8.46}$$

are oblique projectors onto the orthogonal complement $\mathcal{S}_\perp$ of deflation subspace. In particular this means

$$P_X^2 = P_X,\ X = L, R \tag{8.47}$$

$$PP_L = P_R P = 0 \tag{8.48}$$

$$P_L \left(1 - P\right) = \left(1 - P\right) P_R = 1 - P. \tag{8.49}$$

We now consider the component of Dirac equation in the deflation subspace $\mathcal{S}$ by multiplying the linear system from eq. (8.1) by the projector P.

$$\begin{aligned} PAx &= P\left(1 - P_L\right) Ax \\ &= PAx - PAx + \sum_{k,l=1}^{N} PA\phi_k \left(E^{-1}\right)_{kl} \Big(\phi_l, \underbrace{Ax}_{b}\Big) \\ &= \sum_{k,l=1}^{N} PA\phi_k \left(E^{-1}\right)_{kl} \Big(\phi_l, b\Big) \end{aligned} \tag{8.50}$$

We arrive at the explicit solution of the system in the deflation subspace

$$x_{\text{dfl}} = \sum_{k,l=1}^{N} \phi_k \left(E^{-1}\right)_{kl} \Big(\phi_l, b\Big) \tag{8.51}$$

The solution in the orthogonal complement $\mathcal{S}_\perp$ is given implicitly by

$$P_L A x_{\text{compl}} = P_L b. \tag{8.52}$$

We combine the solutions in the two spaces to get

$$x = P_R x_{\text{compl}} + x_{\text{dfl}}. \tag{8.53}$$

Here the component $x_{\text{dfl}}(x)$ in the deflation subspace can be explicitly calculated using the matrix E^{-1}_{kl}. The component in the complement $x_{\text{compl}}(x)$ is the solution to the deflated system in eq. (8.52). The inverse of the deflated operator $\hat{E} = P_L A$ is

$$\hat{E}^{-1} = (1-P)\, E^{-1} \,(1-P)\,. \tag{8.54}$$

If the projector $1-P$ suppresses the low modes of A, the deflated block system is significantly better conditioned and thus faster to invert using the iterative solvers presented in section 8.1.

How to choose and construct the basis vectors of the deflation subspace and accordingly the projectors P and P_L is subject of sections 9.1.1 and 9.2.6. An application of deflated solvers is presented in sections 10.2.2 and 10.3.6.

9 | Domain Decomposition Solvers

The iterative techniques we have seen in the previous chapter are well suited for problems where the Krylov subspace quickly spans a significant part of the overall problem space. If the dimension of the linear system is exceedingly large, very many iterations are needed to span a significant Krylov subspace. After initial rapid convergence, the convergence stagnates. In this case it is useful to divide the original problem into a number of smaller sub problems. Here these sub problems consist of multiple local domains on the lattice. There exist numerous techniques to structure these domains. They differ in the size and shape of the domains and how the equivalence to the original problem is ensured. An overview of several substructuring and domain decomposition techniques is given in [65, 147].

The substructuring into domains serves two main purposes. The control over the subdomain size and resulting dimension of the block problem opens up a wide variety of solver algorithms that are unsuitable for the global problem. In particular, it is possible to use the comparatively simple iterative algorithms presented in chapter 8. In some cases it may even be possible to employ direct, explicit inversion algorithms such as the Gauss elimination. Additionally, the separation into multiple smaller block problems is useful for parallel computer architectures such as computer clusters or graphical processing units (GPU). Acceleration of lattice QCD simulations using GPU architectures is already an active topic [154] and will continue to be relevant in the future. With these parallel architectures in mind, it is beneficial to shift the majority of the workload to the system on the subdomain blocks. This is especially important considering the large local computational power of individual nodes and comparatively slow communication between them.

Here we will introduce two domain decomposition methods. The **S**chwarz **A**lternating **P**rocedure (SAP) will serve as a reference point. It is implemented in the `openQCD` package listed in table A.1 and is used in current lattice QCD simulations [72, 73]. The **F**inite **E**lement **T**ear and **I**nterconnect algorithm (FETI) is the main focus of this work. It is first introduced in [145] and has been refined in [67, 68, 155–159]. Both of these techniques employ rectangular subdomains called blocks. This is a natural choice given the square geometry of the underlying lattice. Although this restriction is not necessary, we consider square blocks of equal size. It is assumed that these blocks evenly divide the global lattice.

In the following we will rephrase the original, global problem in terms of the smaller problems on the subdomains. This is done in several key steps. First we define the block geometry and formulate the individual problems on the blocks. We also define the interaction between different blocks and impose a set of boundary conditions on the blocks. We then establish equivalence between the original problem and the collection of block problems. Finally, we check if the collection of block problems is easier to solve than the original, global problem.

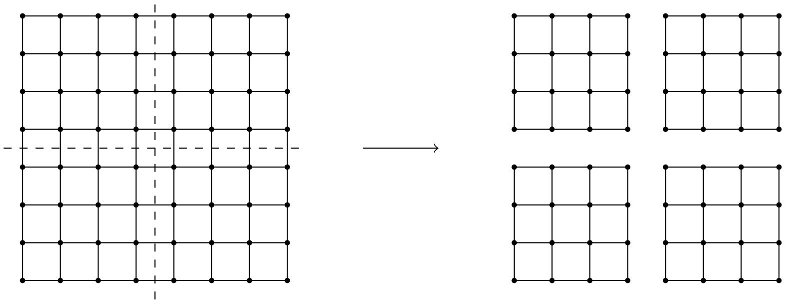


Figure 9.1: Decomposition of global 2d lattice (left) into SAP blocks of side length $s_b = 4$ (right). The individual SAP blocks do not share any points and evenly divide the entire global lattice.

9.1 Schwarz Alternating Procedure

The Schwarz Alternating Procedure (SAP) is used in current QCD applications [72, 73]. It is discussed in this context in [143, 144] and will be used as a reference. A more general overview of the procedure can be found in [147]. The deflation improvement (DFLSAP), which is also part of the `openQCD` package, is presented in [69, 70] and discussed in the next sections.

In the SAP algorithm the lattice is divided into non-overlapping blocks that fill the entire lattice. A 2d schematic of this decomposition is shown in fig. 9.1.

On a given block Λ the operator can be rearranged into

$$A_\Lambda = \begin{pmatrix} A_{\Lambda\Lambda} & A_{\Lambda\Lambda^*} \\ A_{\Lambda^*\Lambda} & A_{\Lambda^*\Lambda^*} \end{pmatrix}, \tag{9.1}$$

where points not in the block Λ are denoted by Λ^*. We have separated the contributions from the block to the block ($\Lambda\Lambda$), the contributions from the outside to the block ($\Lambda\Lambda^*$), the contributions from the block to the outside ($\Lambda^*\Lambda$) and the contributions from the outside to the outside ($\Lambda^*\Lambda^*$). The operator $A_{\Lambda\Lambda}$ is the same as the A except for all terms that involve the boundary fields. These terms are set to zero which is equivalent to imposing Dirichlet conditions on the block boundary.

The SAP algorithm now loops over the blocks sequentially and solves the current block equation. The new approximation to the global solution x' is calculated by solving the following equations

$$Ax' = f, \qquad x \in \Lambda \tag{9.2}$$

$$x' = x, \qquad x \notin \Lambda. \tag{9.3}$$

Combining these two equations the update is given by

$$x' = x + A_{\Lambda\Lambda}^{-1} (f - Ax) \tag{9.4}$$

While this iteration is numerically stable, we can improve it further. If we consider the residue of the new solution

$$\rho' = b - Ax' = \rho - AA_{\Lambda\Lambda}^{-1}\rho \tag{9.5}$$

and assume that the operator A only includes nearest neighbor interactions, we can see that it only differs from ρ on the block Λ itself and all neighboring blocks. This means that every other block can be updated at the same time. If we color the blocks black and white in alternating checkerboard fashion, we can express the update in terms of these two domains only. Black blocks will be labeled Ω and white blocks Ω^*. We can then decompose the operator similarly to eq. (9.1) into the components $A_{\Omega\Omega}, A_{\Omega\Omega^*}, A_{\Omega^*\Omega}$ and $A_{\Omega^*\Omega^*}$. A Schwarz cycle, consists of an update on all black blocks followed by an update on all white blocks and is defined by

$$x \to (1 - KA)\, x + Kb \tag{9.6}$$

$$K = A_{\Omega\Omega}^{-1} + A_{\Omega^*\Omega^*}^{-1} - A_{\Omega^*\Omega^*}^{-1} A_{\Omega^*\Omega} A_{\Omega\Omega}^{-1}. \tag{9.7}$$

After n_{cy} cycles starting from $x = 0$ the solution is

$$x = K \sum_{\nu=0}^{n_{\text{cy}}} (1 - AK)^{\nu}\, b. \tag{9.8}$$

If the SAP algorithm is used as a preconditioner to a global solver, the associated preconditioning operator is

$$M_{\text{SAP}} = K \sum_{\nu=0}^{n_{\text{cy}}} (1 - AK)^{\nu} \tag{9.9}$$

In this case both the number of cycles and the number of iterations for the inversion of the block operators in the definition of K is truncated. In this way we get an approximate to the inverse operator A^{-1} and the preconditioned system is better conditioned.

9.1.1 Deflated Schwarz Alternating Procedure

An overview of the general idea behind deflation techniques was given in section 8.3. Here we will work out how to calculate the deflation subspace $\mathcal{S}$ and define the deflated SAP algorithm as a reference.

In section 8.3 we argued that the basis vectors should be constructed in such a way that $1 - P$ restricts the low modes of the operator. We could therefore choose the global fields $\phi_k,\ k = 1, \cdots, N_s$ to be exactly these low eigenmodes of the operator. However, calculating eigenmodes of the global operator is very computationally expensive. Furthermore, it has been shown in [69] that the knowledge of the exact modes is not needed for efficient deflation of the global system. It is enough to start from a random vector η_k and apply some approximation M of the inverse operator a number ($N_{\text{distillation}}$) of times

$$\phi_k = M^{N_{\text{distillation}}} \eta_k. \tag{9.10}$$

This process is sometimes called distillation and captures the low modes of the system. The approximate versions of the inverse operator can be any preconditioner to the global system. In this case we use the SAP preconditioner introduced in the previous section. While this is a valid deflation scheme, there are further improvements.

We can define a local deflation subspace on blocks Λ^{DFL} similar to the blocks used for the SAP algorithm. It is not necessary to use identical blocks for the SAP and the deflation. The subspace is given by

$$\mathcal{S}_{\Lambda^{\text{DFL}}} = \left\{ \chi_1^{\Lambda^{\text{DFL}}}, \chi_2^{\Lambda^{\text{DFL}}}, \cdots, \chi_{N_s}^{\Lambda^{\text{DFL}}} \right\}. \tag{9.11}$$

The global deflation vectors are the sum of the basis vectors over all blocks.

$$\chi_k = \sum_{\Lambda^{\text{DFL}}} \chi_k^{\Lambda^{\text{DFL}}} \tag{9.12}$$

The construction of these local fields begins with a set of N_s global fields $\phi_k,\ k = 1, \cdots, N_s$. We can choose the approximate low modes of the global operator calculated using the update process in eq. (9.10). These fields are then projected onto the block Λ^{DFL} using the restriction

$$\phi_k^{\Lambda^{\mathrm{DFL}}}(x) = \begin{cases} \phi_k(x), & x \in \Lambda^{\mathrm{DFL}} \\ 0, & \text{otherwise} \end{cases} \tag{9.13}$$

The resulting block fields are then normalized and orthogonalized using the Gram-Schmidt procedure.

$$\left(\chi_1^{\Lambda^{\mathrm{DFL}}}, \chi_2^{\Lambda^{\mathrm{DFL}}}, \cdots, \chi_{N_s}^{\Lambda^{\mathrm{DFL}}}\right) = \text{orthonorm}\left(\phi_1^{\Lambda^{\mathrm{DFL}}}, \phi_2^{\Lambda^{\mathrm{DFL}}}, \cdots, \phi_{N_s}^{\Lambda^{\mathrm{DFL}}}\right) \tag{9.14}$$

This has two major benefits. The application of the projector

$$Px = \sum_{\Lambda^{\mathrm{DFL}}} \sum_{k=1}^{N_s} \phi_k^{\Lambda^{\mathrm{DFL}}} \left(\phi_k^{\Lambda^{\mathrm{DFL}}}, x\right) \tag{9.15}$$

can be executed in parallel on all blocks. More importantly the full deflation subspace is the direct product of all local subspaces. Its dimension is therefore $N = N_b N_s$, where N_b is the number of blocks and N_s is the number of basis vectors on each block. Because of the splitting of the sum in eq. (9.15) the computational cost of the projector is small even for a very high dimensional deflation subspace.

Using the deflation subspace spanned by the basis vectors χ_k one can follow the steps laid out in section 8.3 and solve the deflated system. It is, however, possible to construct a more efficient algorithm, by using the SAP and deflation techniques as a preconditioner to the global GCR algorithm [64, 69, 70].

9.1.2 Deflated SAP as a Preconditioner

The deflated SAP (DFLSAP) preconditioner [64, 69, 70] consists of two steps. In the first step we use the deflation subspace to construct a preconditioner that approximates the low modes of the global operator A. We then employ a small number of steps of the SAP algorithm to estimate the smaller details. To combine the two preconditioners we first approximate the solution using the deflation subspace

$$x_1 = M_{\mathrm{DFL}} f = (PAP)^{-1} f, \tag{9.16}$$

The deflation preconditioner M_{DFL} is an approximation to the inverse of the little operator PAP. The little system is much easier to solve given its smaller size and better condition. The full solution is now given by the approximation x_1 and the remainder x_2

$$A(x_1 + x_2) = f. \tag{9.17}$$

The remainder x_2 is implicitly defined by

$$Ax_2 = f - M_{\mathrm{DFL}} f. \tag{9.18}$$

We can now iteratively improve the solution x_1 by approximating the solution to this second residual system. For the second approximation we use the SAP preconditioner defined in eq. (9.9) to get

$$x_2 \approx M_{\mathrm{SAP}} \left(f - M_{\mathrm{DFL}} f\right). \tag{9.19}$$

Combining the two components we get

$$x = x_1 + x_2 \approx \left(M_{\mathrm{DFL}} + M_{\mathrm{SAP}} - M_{\mathrm{SAP}} A M_{\mathrm{DFL}}\right) f. \tag{9.20}$$

This defines the preconditioner

$$M_{\mathrm{DFLSAP}} = M_{\mathrm{DFL}} + M_{\mathrm{SAP}} - M_{\mathrm{SAP}} A M_{\mathrm{DFL}} \tag{9.21}$$

combining the SAP and deflation techniques.

9.2 Finite Element Tear Interconnect Algorithm

In the **F**inite **E**lement **T**ear and **I**nterconnect algorithm [65, 145] the global lattice is also split into blocks. In contrast to the SAP, the points on the interfaces of the blocks are duplicated and attributed to every connected block. While this increases the size of the overall problem, the individual blocks are now "more independent" of each other. Since the interaction decays exponentially in the distance, points far away from block boundaries can be accurately solved without the need for information from the neighboring blocks. This is an advantage for parallel computer architectures, where a large amount of computational power is available, but the communication between different processors is comparatively slow. Of course the individual blocks are not completely independent of each other. For this reason we have to enforce additional constraints that govern the interaction between blocks. This step extends the overall problem size again. However, the possibility to parallelize in a more efficient way may outweigh these additional costs.

We will now examine the FETI algorithm. The geometry of the algorithm and its blocking will be defined in section 9.2.1. In section 9.2.2 we consider the interaction between different blocks and the continuity of the solution across block boundaries. We will then use the definitions of the previous sections in section 9.2.4 to define block versions of the involved operators and relate them back to their counterparts on the global lattice. Different preconditioning techniques and improvements to the FETI algorithm are presented in sections 9.2.6 and 9.2.7.

9.2.1 Lattice Geometry

To illustrate the lattice geometry we are using a 2d schematic of the lattice. Wherever the generalization to higher dimensions is not straight forward, we will go into detail explicitly. The left side of fig. 9.2 shows the global lattice. A spinor resides on each site of the lattice and gauge links connect neighboring sites, i.e. sites that are one lattice spacing apart. The global vector containing the spinors on all sites of the global lattice is called x_g. In the example shown we also see a division into 4 subdomains or blocks. The lattice points on the interior or bulk of the blocks are shown as points, while points on the boundary are shown as triangles or squares for edges and corners respectively. When we formulate the problem in terms of block quantities, it will be useful to divide the lattice into the space of interior (bulk) variables x_I and boundary variables x_Δ. The block sizes are chosen such that the domains overlap by exactly one lattice unit on the edge. As a consequence the variables on the boundary are duplicated and exist separately on each connected block.

It will be helpful to define a vector that more closely resembles this sublattice structure. The vector $x_{\{b\}}$ is a collection of all the block vectors x_b^α. Each block vector is arranged such that the bulk variables are followed by the boundary variables.

$$\begin{aligned} x_{\{b\}} &= \left(x_b^1, x_b^2, \cdots, x_b^N\right)^T \\ &= \left(\left(x_I^1, x_\Delta^1\right)^T, \left(x_I^2, x_\Delta^2\right)^T, \cdots, \left(x_I^N, x_\Delta^N\right)^T\right)^T \end{aligned} \tag{9.22}$$

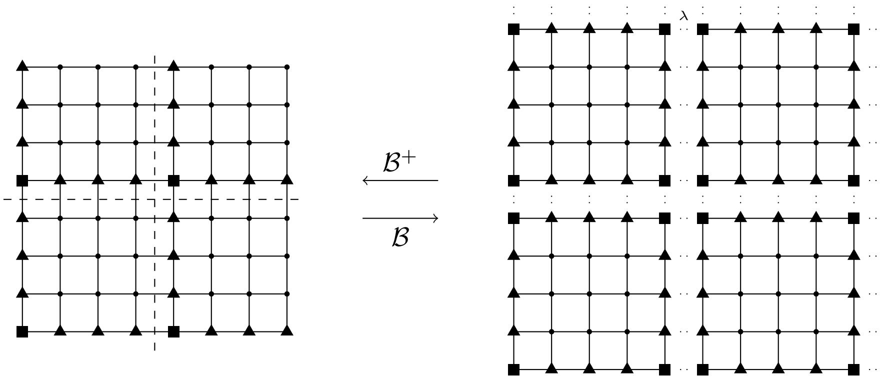


Figure 9.2: Domain decomposition of a 2d global lattice (left) into FETI blocks (right). The operators $\mathcal{B}$ and $\mathcal{B}^+$ relate between the two representations. Lattice points on the interior of the blocks are shown as points. The boundaries are shown as triangles on the edges and as squares on the corners. Note that the boundary points are duplicated on neighboring blocks. Boundary points that correspond to the same point on the global lattice are connected by dotted lines indicating the Lagrange multipliers.

To relate block representation and the global representations, we use an operator B^α that projects the global vector x_g onto the block α.

$$x_b^\alpha = \begin{pmatrix} x_I^\alpha \\ x_\Delta^\alpha \end{pmatrix} = \begin{pmatrix} B_I^\alpha \\ B_\Delta^\alpha \end{pmatrix} x_g = B^\alpha x_g \tag{9.23}$$

The operators B_I^α and B_Δ^α project into the bulk and boundary subspace on the block α respectively. The collection of all block projectors B^α is called $\mathcal{B}$ and acts as

$$x_{\{b\}} = \begin{pmatrix} B^1 \\ \\ B^2 \\ \\ \vdots \\ B^N \\ \\ \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_V \end{pmatrix} = \mathcal{B} x_g. \tag{9.24}$$

The operator $\mathcal{B}$ has dimension $(N_B V_B) \times V_g$, with N_B and V_B being the number of blocks and their volume and V_g the volume of the global lattice. It transports the lattice points from the global lattice onto the individual blocks. It consists purely of (positive) ones and zeros in the appropriate places. An explicit example of these block operators for a very small lattice is given in section 9.2.3.

We also define operators for the inverse step - collecting variables from the blocks onto the global lattice.

$$x_g = \mathcal{B}^+ x_{\{b\}} \tag{9.25}$$

Since $\mathcal{B}$ is singular we can only define the pseudo-inverse $\mathcal{B}^+$. To find one possible pseudo-inverse, we follow different lattice points as they are projected to the block representation and back. Let us first consider a point in the bulk of a block. There is a one to one correspondence of points in the interior of a block and points on the global lattice. The block projector for bulk points is therefore similar to the unit matrix with possible rows and columns rearranged (See eq. (9.35) for an explicit example). Accordingly, the inverse is the transposed matrix

$$B_I^+ = B_I^{-1} = B_I^T. \tag{9.26}$$

Points on the boundary are transferred to multiple blocks. The transposed of the block projector sums the contributions from the different block boundaries that correspond to the same global lattice point. The pseudo-inverse needs to correct this double counting with a weight matrix W_Δ.

$$B_\Delta^+ = B_\Delta^T W_\Delta \tag{9.27}$$

When choosing this weight matrix there is some freedom. We need to ensure that the sum of the weights of boundary points that correspond to the same global lattice point is one. This means

$$\mathcal{B}^+\mathcal{B} = \mathcal{B}^T W \mathcal{B} = \mathbb{1} \tag{9.28}$$

implicitly defining the weight matrix W. There is some freedom in choosing the weight matrix that will be discussed briefly in section 9.2.7. Here we use equal weights for each boundary type. Points on block edges are transferred to exactly two connected blocks. Hence, they receive a weight of $\frac{1}{2}$. Points on corners contribute to 4, 8 and 16 blocks and are weighted with $\frac{1}{4}$, $\frac{1}{8}$ and $\frac{1}{16}$ respectively. A concrete example of the weight matrix is given in section 9.2.3.

9.2.2 Continuity Conditions

Through the application of the $\mathcal{B}$ operator we have transferred the original problem

$$A\,x = f \tag{9.29}$$

into a system of overlapping, independent block equations

$$A_{\{b\}}\mathcal{B}x = \mathcal{B}f \Leftrightarrow \tag{9.30}$$

$$\begin{pmatrix} A_b^1 & 0 & \cdots & 0 \\ 0 & A_b^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_b^N \end{pmatrix} \begin{pmatrix} x_b^1 \\ x_b^2 \\ \vdots \\ x_b^N \end{pmatrix} = \begin{pmatrix} f_b^1 \\ f_b^2 \\ \vdots \\ f_b^N \end{pmatrix}. \tag{9.31}$$

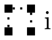
Ultimately we want to solve the block system because its shape is beneficial to parallel architecture. The block operators A_b^α and their relation to the original operator A will be properly defined in section 9.2.4. Note that since we doubled the points on block boundaries the system is not (yet) equivalent to the original system. We need to fix the additional degrees of freedom introduced by the doubling of the boundaries. To do this we require block points that correspond to the same global lattice point to be equal. The condition reads

$$x_\Delta^\alpha - x_\Delta^\beta = 0 \tag{9.32}$$

for points on the boundary of neighboring blocks α and β that correspond to the same global lattice point. The solution is then continuous across block boundaries. These additional constraints are added to the system using a set of Lagrange multipliers λ. The continuity constraints are visualized in figs. 9.2 and 9.3 as dotted lines between the blocks that link connected boundary points. There is some ambiguity in the distribution of the sign. We choose a checkerboard pattern such that the boundaries on even blocks receive a negative sign while boundaries on odd

	1d	2d	3d	4d
no. points	2	4	8	16
no. constraints	1	4	12	32
rank (C_Δ)	1	3	7	15

Table 9.1: Constrains used for the four corner types. We give the number of points involved in a corner and the number of constraints or Lagrange multipliers λ used. To show the redundancy of the constraints we also list the rank of the matrix used to encode the constraints.

blocks stay positive. On corner points more than two blocks are connected. There are multiple ways to connect these points, some of which lead to redundant constraints. For a 2d corner the pattern  is enough to constrain all four corner points. As in fig. 9.2 dotted line represent Lagrange multipliers. To keep the symmetry and for convenience in the implementation we choose to constrain all next neighbors, i.e. the pattern , including the redundant constraint at the bottom. Note that no diagonal constraints are used. For higher dimensional corners we also use the redundant constraints that connect all next neighbors.

The conditions of the form in eq. (9.32) are encoded in the matrix $\mathcal{C}$ that act on the boundaries of blocks. The matrix $\mathcal{C}$ has dimension $(N_B V_B) \times V_\lambda$, where V_λ is the number of Lagrange multipliers. It is a collection of block matrices.

$$\mathcal{C} = \begin{pmatrix} C^1 \\ C^2 \\ \vdots \\ C_N \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} 0 \\ C^1_\Delta \end{pmatrix} \\ \begin{pmatrix} 0 \\ C^2_\Delta \end{pmatrix} \\ \vdots \\ \begin{pmatrix} 0 \\ C^N_\Delta \end{pmatrix} \end{pmatrix} \tag{9.33}$$

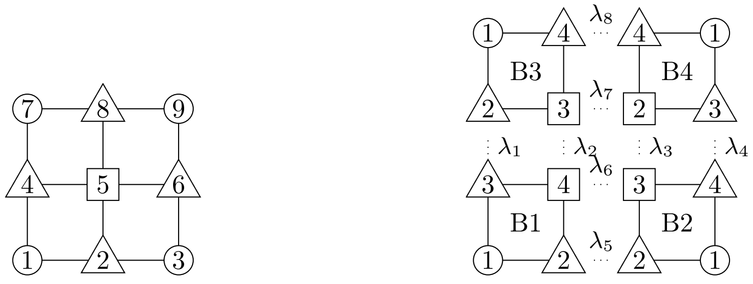
Note that the block matrices C^α only act on the boundary part. The matrices consist of mostly 0 and ± 1 for points on different blocks that correspond to the same global lattice point. In table 9.1 we list some characteristics of the constraints. We list the number of points involved in this boundary and the number of constraints used to tie them together. These numbers govern the size of the matrix C_Δ. To illustrate the redundant constraints, we also list the rank of C_Δ, which is the minimum number of constraints needed. An example of the continuity matrices is given in section 9.2.3.

Only by adding these constraints to the block system in eq. (9.31) it becomes equivalent to the original system in eq. (9.29). We extend the block system to encompass the original solution x as well as the Lagrange multipliers λ.

$$\left(\begin{array}{cccc|c} A^1_b & 0 & \cdots & 0 & \left(C^1\right)^T \\ 0 & A^2_b & \cdots & 0 & \left(C^2\right)^T \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & A^N_b & \left(C^N\right)^T \\ \hline C^1 & C^2 & \cdots & C^N & 0 \end{array}\right) \left(\begin{array}{c} x^1_b \\ x^2_b \\ \vdots \\ x^N_b \\ \hline \lambda \end{array}\right) = \left(\begin{array}{c} f^1_b \\ f^2_b \\ \vdots \\ f^N_b \\ \hline 0 \end{array}\right). \tag{9.34}$$

This system will be discussed in detail in section 9.2.5.

In the following section we want to broaden the understanding of the lattice geometry and the projection and continuity operators by writing them out in explicit form.



(a) Global representation.

(b) Block representation.

Figure 9.3: Schematic of the global (left) and block (right) representation of the lattice. The numbers show the ordering of lattice points. The right hand side shows the four blocks labeled B1 to B4. The Lagrange multipliers are shown as dotted lines connecting neighboring points. Their ordering is also defined. The operator $\mathcal{B}$ is used to transfer points from the global lattice to the blocks. The (pseudo)inverse operator $\mathcal{B}^+$ transfers points from the block representation to the global lattice, averaging block points that correspond to the same global point.

9.2.3 An Explicit Example

The operators described in the sections above are large for any meaningful lattice sizes. For this reason they are rarely given explicitly. To illustrate the global and block representations we will consider a tiny 2d 3×3 lattice divided into four blocks. The left side in fig. 9.3 shows the global representation of the lattice. The right side shows the decomposition into four blocks labelled B1 to B4 containing four lattice points each.[1] Note that the edges $(2, 4, 6, 8)$ drawn as triangles are shared between two blocks each and the corner (5) drawn as a square is part of all four blocks. The block projection operator $\mathcal{B}$ has the form

$$\mathcal{B} = \begin{pmatrix} B^1 \\ B^2 \\ B^3 \\ B^4 \end{pmatrix} = \left(\begin{array}{ccccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right) . \tag{9.35}$$

The transposed of this matrix sums over 2 points for edges (columns 2,4,6 and 8) and over 4 points for the corner point 5. The weight matrix must correct this double counting. It is applied

[1] Note that later the actual shape of the blocks will be different. The simple example here is only used to illustrate the block projection operators.

in the space of the block variables and has the form

$$W = \operatorname{diag}\left(\begin{array}{cccc|cccc|cccc|cccc} 1 & \tfrac{1}{2} & \tfrac{1}{2} & \tfrac{1}{4} & 1 & \tfrac{1}{2} & \tfrac{1}{4} & \tfrac{1}{2} & 1 & \tfrac{1}{2} & \tfrac{1}{4} & \tfrac{1}{2} & 1 & \tfrac{1}{4} & \tfrac{1}{2} & \tfrac{1}{2} \end{array}\right). \tag{9.36}$$

The weight matrix is used in the definition of $\mathcal{B}^+$ that collects points from the individual blocks back to the global lattice.

$$\mathcal{B}^+ = \mathcal{B}^T W = \left(\begin{array}{cccc|cccc|cccc|cccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \tfrac{1}{2} & 0 & 0 & 0 & \tfrac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \tfrac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & \tfrac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \tfrac{1}{4} & 0 & 0 & \tfrac{1}{4} & 0 & 0 & 0 & \tfrac{1}{4} & 0 & 0 & \tfrac{1}{4} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \tfrac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & \tfrac{1}{2} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \tfrac{1}{2} & 0 & 0 & 0 & \tfrac{1}{2} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{array}\right). \tag{9.37}$$

We can confirm that transferring the points from the global lattice to the blocks using $\mathcal{B}$ and back using $\mathcal{B}^+$ restores the original configuration ($\mathcal{B}^+\mathcal{B} = \mathbb{1}$). The weight matrix W cancels the summation of points that are transferred to multiple block boundaries.

The continuity matrix relates pairs of points on the boundary of neighboring blocks to the Lagrange multipliers λ. The sign is distributed in a checkerboard pattern. Even blocks receive a negative sign. The continuity matrix $\mathcal{C}$ is a collection of block continuity matrices and reads as follows.

$$\begin{aligned}\mathcal{C} &= \begin{pmatrix} C^1 & C^2 & C^3 & C^4 \end{pmatrix} \\ &= \left(\begin{array}{cccc|cccc|cccc|cccc}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1
\end{array}\right)\end{aligned} \tag{9.38}$$

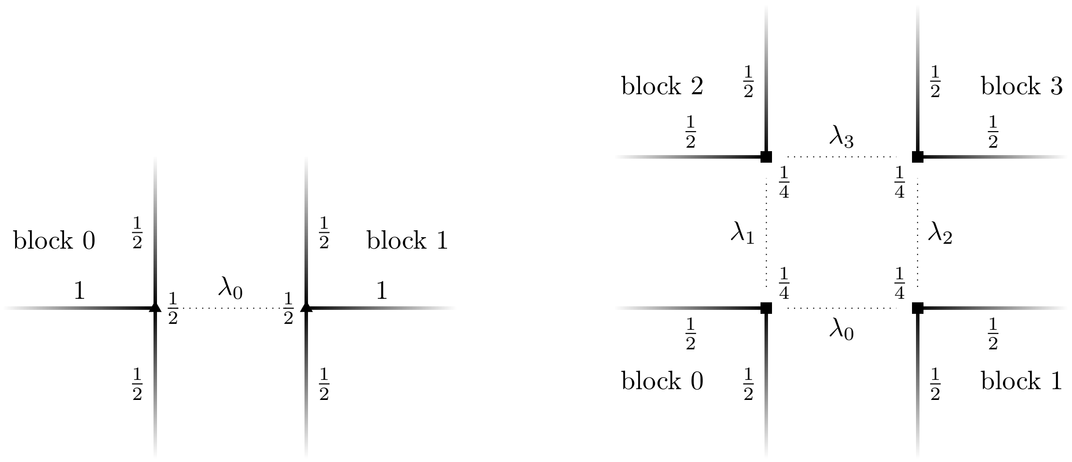
Each row connects boundary points in two blocks with a relative sign. Columns corresponding to the interior of the blocks (1,4,8,12) are empty. The corner point is connected through the Lagrange multipliers $\lambda_2, \lambda_3, \lambda_6$ and λ_7. One of the multipliers is redundant. This can be seen from the matrix since each of the rows 2,3,6 and 7 is a linear combination of the others.

9.2.4 Block Quantities

So far we have seen how to project the original system onto the individual FETI blocks. We also defined how these blocks are tied together using the continuity constraints and the Lagrange multipliers λ. In this section we will establish how we have to treat the block operators such that the FETI system is equivalent to the original system. For clarity, we will focus on the two-dimensional operator first and then generalize to the four-dimensional version.

In two dimensions the next neighbor operator connects each point to itself as well as the four neighboring points. We can write it as

$$A_{xy} = S_x \delta_{x,y} + L_{x,\mathcal{N}(x)} \delta_{\mathcal{N}(x),y} \tag{9.39}$$



(a) One dimensional interface (edge) with one lattice point represented on two blocks.

(b) Two dimensional interface (corner) with one point represented on four blocks.

Figure 9.4: Interface points for FETI blocks. One point on the global lattice is duplicated onto the boundary of multiple blocks. We give the weights W_{S_x} on the site x next to the points as well as the weights $W_{L_{x,\mathcal{N}(x)}}$ on the links.

where we have split the operator into the part S_x attributed to the site at x and the part $L_{x,\mathcal{N}(x)}$ attributed to the link connecting the site x to its neighbors $\mathcal{N}(x)$. We require that the block system is equivalent to the global system once it is transformed back using to the global space using the operator $\mathcal{B}$ defined in section 9.2.1.

$$\mathcal{B}^T A^b = A^g \tag{9.40}$$

If a point is in the interior of a block (I), all of its neighbors are also on the same block. The definition of the block operator is thus the same as for the global operator.

$$\left(A^b_{IX}\right)_{xy} = A^g_{xy}, \quad X = I, \Delta \tag{9.41}$$

If a point is on the block boundary (Δ), however, the definition must change since not all neighboring sites can be reached. Additionally, when transforming the block system back to the global lattice, the contributions from block boundaries get summed, according to the definition of $\mathcal{B}^T$ in section 9.2.1. Because of this summation of boundary points we have to apply weights inside the operator to avoid double counting. We will have weights W_{S_x} for the contribution S_x coming from the site x and weights $W_{L_{x,\mathcal{N}(x)}}$ for the contribution $L_{x,\mathcal{N}(x)}$ that stems from the link connecting the site x to its neighbors. For boundary points we therefore get

$$\left(A^b_{\Delta X}\right)_{xy} = W_{S_x} S_x \delta_{x,y} + W_{L_{x,\mathcal{N}(x)}} L_{x,\mathcal{N}(x)} \delta_{\mathcal{N}(x),y}, \quad X = I, \Delta \tag{9.42}$$

The weights W_{S_x} and $W_{L_{x,\mathcal{N}(x)}}$ depend on the type of boundary the points x and $\mathcal{N}(x)$ are located on. The weight of a link that sticks outside the block is zero. We will work out the weights for the other cases in the following for each boundary type individually. First we consider two connected points on neighboring blocks. Figure 9.4a depicts a schematic of such a 2d boundary point pair and the two blocks is it attributed to. We also show the links that connect it to the neighboring sites. The contributions from the left (right) link only appear in block 0 (block 1).

The contributions from the top and bottom links are counted on each block. They receive a weight such that they are only counted once when transferred back to the global lattice. We are free to choose the weights arbitrarily as long as their sum is one. We decided on a scheme, where the weights are equal. As a result we apply a weight of $W_{L_{x,\mathcal{N}(x)}} = \frac{1}{2}$ to the top and bottom links on the two blocks. The contribution from the site itself also gets a weight of $W^{\text{1d}}_{S_x} = \frac{1}{2}$. These weights are also represented in fig. 9.4a.

Next we consider a 2d corner. A schematic of this corner type is shown in fig. 9.4b. Here the point gets duplicated onto four blocks. Accordingly, the weight on the sites must be $W^{\text{2d}}_{S_x} = \frac{1}{4}$. All neighboring points are on the edge and get copied to two blocks each. It follows that the links connecting these points exist on two blocks each as well. The link weight is $W_{L_{x,\mathcal{N}(x)}} = \frac{1}{2}$ for all involved links.

For three and four dimensional corners the site weight is $W^{\text{3d}}_{S_x} = \frac{1}{8}$ and $W^{\text{4d}}_{S_x} = \frac{1}{16}$ respectively. The weight of the link depends on the number of blocks where this contribution is counted, i.e. the number of blocks the neighboring point appears in. It can therefore be related to the site weight two points connected by the link. For any link the weight is

$$W_{L_{x,\mathcal{N}(x)}} = \max\left(W_{S_x}, W_{S_{\mathcal{N}(x)}}\right). \tag{9.43}$$

These weights add factors of $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}$ and $\frac{1}{16}$ to some diagonal entries of the original operator and factors $\frac{1}{2}, \frac{1}{4}$ and $\frac{1}{8}$ to the off diagonal entries that correspond to boundary terms. This affects the condition number of the block operators, which will in turn affect the convergence of iterative solvers. Compared with the unweighted block operator the condition number is increased. Whether this problem can be overcome by the benefits of the FETI algorithm will be topic of section 10.3, where we give an analysis on the convergence of the block system. There we will discuss preconditioning and deflation techniques for the block system.

9.2.5 FETI System

Sections 9.2.1 and 9.2.2 lead to the FETI system in eq. (9.34). In section 9.2.4 we have defined the block operators and established the equivalence of the FETI system and the original system. We will repeat the system here using the explicit division into interior (I) and boundary (Δ) components.

$$\left(\begin{array}{ccccccc|c}
A^1_{II} & A^1_{I\Delta} & 0 & 0 & \cdots & 0 & 0 & 0 \\
A^1_{\Delta I} & A^1_{\Delta\Delta} & 0 & 0 & \cdots & 0 & 0 & \left(C^1_\Delta\right)^T \\
0 & 0 & A^2_{II} & A^2_{I\Delta} & \cdots & 0 & 0 & 0 \\
0 & 0 & A^2_{\Delta I} & A^2_{\Delta\Delta} & \cdots & 0 & 0 & \left(C^2_\Delta\right)^T \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & A^N_{II} & A^N_{I\Delta} & 0 \\
0 & 0 & 0 & 0 & \cdots & A^N_{\Delta I} & A^N_{\Delta\Delta} & \left(C^N_\Delta\right)^T \\
\hline
0 & C^1_\Delta & 0 & C^2_\Delta & \cdots & 0 & C^N_\Delta & 0
\end{array}\right)
\left(\begin{array}{c}
x^1_{b,I} \\ x^1_{b,\Delta} \\ x^2_{b,I} \\ x^2_{b,\Delta} \\ \vdots \\ x^N_{b,I} \\ x^N_{b,\Delta} \\ \hline \lambda
\end{array}\right)
=
\left(\begin{array}{c}
f^1_{b,I} \\ f^1_{b,\Delta} \\ f^2_{b,I} \\ f^2_{b,\Delta} \\ \vdots \\ f^N_{b,I} \\ f^N_{b,\Delta} \\ \hline 0
\end{array}\right) \tag{9.44}$$

The upper left part of the matrix holds the N individual block systems. Each block point has contributions from other points on the same block. The boundary points also have contributions from neighboring blocks through the Lagrange multipliers and the matrices $(C^\alpha_\Delta)^T$. The last line consists of the continuity constraints C^α_Δ that tie neighboring blocks together and fix the degrees of freedom added when duplicating the block boundaries.

We will now write the system in a more compact form by using the collection of all block operators for a specific domain.

$$\left(\begin{array}{cc|c} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} & 0 \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} & \mathcal{C}_{\Delta}^{T} \\ \hline 0 & \mathcal{C}_{\Delta} & 0 \end{array}\right) \left(\begin{array}{c} x_I \\ x_\Delta \\ \hline \lambda \end{array}\right) = \left(\begin{array}{c} f_I \\ f_\Delta \\ \hline 0 \end{array}\right) \tag{9.45}$$

The vectors x_X with $X = I, \Delta$ include components on all blocks

$$x_X = \left(\begin{array}{cccc} x_X^1 & x_X^2 & \cdots & x_X^N \end{array} \right)^T . \tag{9.46}$$

The operators $\mathcal{A}_{XY}$ are defined accordingly as the collection of block operators

$$\mathcal{A}_{XY} = \begin{pmatrix} A_{XY}^1 & 0 & \cdots & 0 \\ 0 & A_{XY}^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{XY}^N \end{pmatrix} . \tag{9.47}$$

Using the compact notation from eq. (9.45) we eliminate the variables x_I and x_Δ. Splitting the first two rows and the last row we get

$$\begin{pmatrix} x_I \\ x_\Delta \end{pmatrix} = \begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix}^{-1} \left[\begin{pmatrix} f_I \\ f_\Delta \end{pmatrix} - \begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix} \lambda \right] \tag{9.48}$$

$$\begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} x_I \\ x_\Delta \end{pmatrix} = 0. \tag{9.49}$$

Inserting we arrive at an equation for λ

$$\begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix}^{-1} \left[\begin{pmatrix} f_I \\ f_\Delta \end{pmatrix} - \begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix} \lambda \right] = 0. \tag{9.50}$$

Rearranging yields a system of equations for the Lagrange multipliers λ

$$\left[\begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix} \right] \lambda = \begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix}^{-1} \begin{pmatrix} f_I \\ f_\Delta \end{pmatrix} \tag{9.51}$$

$$F_{\lambda\lambda} \lambda = \tilde{f}. \tag{9.52}$$

Here we have defined the quantities

$$F_{\lambda\lambda} = \begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix} \tag{9.53}$$

$$\tilde{f} = \begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix}^{-1} \begin{pmatrix} f_I \\ f_\Delta \end{pmatrix} . \tag{9.54}$$

In eq. (9.45) we have a global system of equations for the Lagrange multipliers λ. Once these values are known, the solution x can be recovered on each block independently using eq. (9.48).

Summarizing, the FETI algorithm consists of the following steps

1. Preparation of the right-hand side $\tilde{f}$ using eq. (9.54)
2. Solution of the FETI-system of the Lagrange multipliers given in eq. (9.52)
3. Resubstitution using the Lagrange multipliers and eq. (9.48).

9.2.6 Preconditioners for the FETI Algorithm

General preconditioning techniques were discussed in section 8.2. In this section we will work on step 2 of the FETI algorithm listed above. We will present two commonly used preconditioners for the FETI system [65, 155] that approximate the inverse of the FETI system $F_{\lambda\lambda}^{-1}$. How these preconditioners compare in our QCD application is discussed in sections 10.2.1 and 10.2.2.

Lumped Preconditioner

The Lumped preconditioner [145] works by taking advantage of the inverse in eq. (9.53). We want to approximate the inverse

$$F_{\lambda\lambda}^{-1} = \left(\begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix} \right)^{-1} . \tag{9.55}$$

Noting the nested inverse we can approximate the operator by neglecting the inversion of the continuity matrices C_Δ. We arrive at the Lumped preconditioner

$$M_{\mathrm{LPD}} = \begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix} \begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix} . \tag{9.56}$$

Dirichlet Preconditioner

The Dirichlet preconditioner [145, 160] works similarly to the Lumped preconditioner seen in the previous section. We also neglect the inversion of the continuity matrices C_Δ. However, we express the inverse of the block operator as

$$\begin{pmatrix} \mathcal{A}_{II} & \mathcal{A}_{I\Delta} \\ \mathcal{A}_{\Delta I} & \mathcal{A}_{\Delta\Delta} \end{pmatrix}^{-1} = \begin{pmatrix} \mathcal{S}_{II}^{-1} & -\mathcal{S}_{II}^{-1}\mathcal{A}_{I\Delta}\mathcal{A}_{\Delta\Delta}^{-1} \\ -\mathcal{A}_{\Delta\Delta}^{-1}\mathcal{A}_{\Delta I}\mathcal{S}_{II}^{-1} & \mathcal{S}_{\Delta\Delta}^{-1} \end{pmatrix} \tag{9.57}$$

using the Schur complements

$$\mathcal{S}_{II} = \mathcal{A}_{II} - \mathcal{A}_{I\Delta}\mathcal{A}_{\Delta\Delta}^{-1}\mathcal{A}_{\Delta I} \tag{9.58}$$

$$\mathcal{S}_{\Delta\Delta} = \mathcal{A}_{\Delta\Delta} - \mathcal{A}_{\Delta I}\mathcal{A}_{II}^{-1}\mathcal{A}_{I\Delta}. \tag{9.59}$$

If we further note, that the continuity constraints only act on the boundary components,

$$\begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & \mathbb{1}_\Delta \end{pmatrix} \begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix}, \tag{9.60}$$

we can define the Dirichlet preconditioner as

$$M_{\mathrm{DIR}} = \begin{pmatrix} 0 & \mathcal{C}_\Delta \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & \mathcal{S}_{\Delta\Delta} \end{pmatrix} \begin{pmatrix} 0 \\ \mathcal{C}_\Delta^T \end{pmatrix} . \tag{9.61}$$

Eigenvalue Preconditioner

To deflate the FETI system we adapt the techniques introduced for the deflated SAP in [69, 70] and section 9.1.1. In the deflated SAP the local deflation basis is constructed on the blocks. The FETI operator $F_{\lambda\lambda}$ acts in the space of the Lagrangian multipliers and therefore connects the boundary faces of neighboring blocks. Accordingly, we define the deflation vectors on the individual block faces.

Similar to the deflated SAP algorithm (section 9.1.1), we first calculate global modes using the repeated application of one of the preconditioners. Here we use the Dirichlet preconditioner described in the previous section. In this way we approximate the low eigenmodes of the $F_{\lambda\lambda}$ operator. From these global modes ϕ_k we construct a set of local modes $\chi_k^{\Lambda_\Delta}$. The index Λ_Δ labels the individual boundary faces on the block Λ. We construct the local deflation basis by projecting the global modes onto the individual faces

$$\phi_k^{\Lambda_\Delta} = \begin{cases} \phi_k, & x \in \Lambda_\Delta \\ 0, & \text{otherwise.} \end{cases} \tag{9.62}$$

We then normalize and orthogonalize using the Gram-Schmidt method

$$\left(\chi_1^{\Lambda_\Delta}, \chi_2^{\Lambda_\Delta}, \cdots, \chi_{N_s}^{\Lambda_\Delta}\right) = \text{orthonorm}\left(\phi_1^{\Lambda_\Delta}, \phi_2^{\Lambda_\Delta}, \cdots, \phi_{N_s}^{\Lambda_\Delta}\right) \tag{9.63}$$

to get the deflation vectors $\chi_k^{\Lambda_\Delta}$. Using these local modes on the block faces we can define the projector

$$P = \sum_{\Lambda_\Delta} \sum_k \chi_k^{\Lambda_\Delta} \left(\chi_k^{\Lambda_\Delta}\right)^\dagger \tag{9.64}$$

From here we proceed in the same way as for the deflated SAP preconditioner presented in section 9.1.2. We solve the little system to get a first approximate solution. This defines the deflation preconditioner for the FETI system

$$M_{\lambda\text{DFL}} = \left(P F_{\lambda\lambda} P\right)^{-1}. \tag{9.65}$$

We then improve the approximation using the Dirichlet preconditioner. The resulting combined preconditioner is defined by

$$M_{\text{DFLFETI}} = M_{\lambda\text{DFL}} + M_{\text{DIR}} - M_{\text{DIR}} F_{\lambda\lambda} M_{\lambda\text{DFL}}. \tag{9.66}$$

9.2.7 Improvements to the FETI Algorithm

The literature [65, 67, 68, 156, 157, 161] contains a variety of improvements for the FETI algorithm. Some of them will be briefly mentioned here.

FETI-DP

In section 9.2 we presented the regular FETI algorithm. A common extension to the FETI algorithm is the dual-primal FETI algorithm (FETI-DP) [65, 67, 156]. The FETI algorithm used here distinguishes between points in the interior of blocks and points on the boundary. In the FETI-DP algorithm a new subspace for the corner points is created. The new geometry is shown in fig. 9.5. This primal subspace is then inverted globally. This is feasible if the primal subspace is substantially smaller than the global problem. It serves as a way to quickly shape the global, coarse structure of the solution, while the finer details are solved on the individual blocks. It also solves the issue of redundant constraints encountered in section 9.2.2 since primal points are global and therefore not part of the continuity constraints.

In preliminary studies in two dimensions we successfully implemented this algorithm. However, in four dimensions it is unclear which of the corner types to include in the primary subspace. If we include all corners, the primary system becomes very large. On top of that including all corners means that a lot of the primary points are nearest neighbors. The distinction between the coarse primary and the fine block systems is no longer given. The primary system would contribute substantially to the solution and the benefits are reduced. For these reasons we chose not to implement the FETI-DP algorithm in 4d QCD.

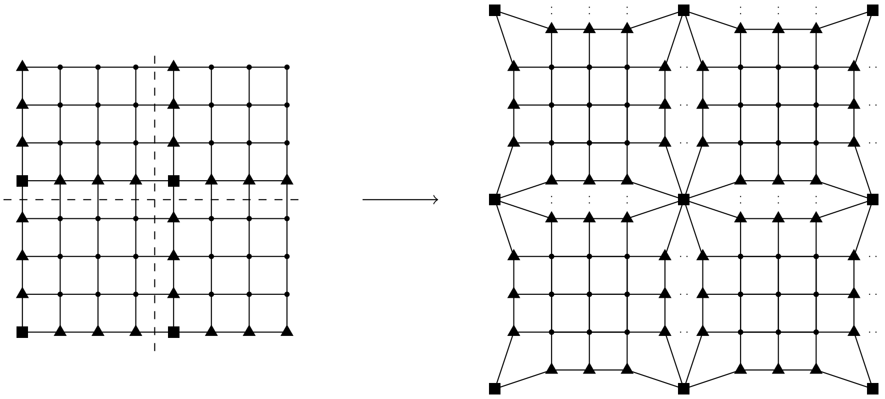


Figure 9.5: Domain decomposition of a 2d global lattice (left) into FETI-DP blocks (right). Interior lattice points are shown as points. Lattice points on the edge are shown as triangles and duplicated on the blocks. Corner points are represented by squares and are attributed to a new primal space. They are not part of the blocks.

Scaling

In section 9.2.1 we introduced the geometry of the FETI algorithm. We introduced the operators that project the lattice points to and from the blocks and the weight matrix that scales boundary points on the blocks. We require the sum over all weights that correspond to the same global lattice point to be one. Similarly, in section 9.2.4 we introduced weights in the block operators to ensure equivalence between the FETI system and the original system. We use a single weight for each boundary type that corresponds to its inverse multiplicity. There are methods which alter these weights on order to optimize the condition of the operators [67, 68, 161]. For this initial consideration of the FETI algorithm in 4d lattice QCD we chose not to implement these scaling techniques.

10 | FETI in Lattice QCD

In this section we will consider the application of the FETI algorithm in lattice QCD. The lattice characteristics and geometry are laid out in chapter 1. In chapter 9, where we introduced the algorithms, we focused on the more illustrative two-dimensional systems. Here we implement the four dimensional versions of the algorithms.

We will switch from the notation that is frequently used when discussing iterative solvers to one that is more common for lattice QCD. So far, the system of equations we want to solve is defined by the matrix A which was left unspecified. In lattice QCD the equation of interest is the Dirac equation

$$D\psi(x) = \eta(y) \tag{10.1}$$

given by the Dirac operator defined in eq. (1.30). The right-hand side $\eta(y)$ and the solution vector $\psi(x)$ are spinors on lattice points y and x each containing 24 degrees of freedom (see chapter 1). The points x and y are part of a four dimensional lattice. Spinors on the 8 neighboring lattice sites are connected through gauge links made up of SU(3) matrices.

Earlier applications of the FETI algorithm are in two dimensions [66, 67, 155, 157, 162]. Only more recently have there been results in three dimensions [158, 161, 163]. Here we analyze in what way the FETI algorithm can be generalized to four dimensions. Additionally, we compare the FETI algorithm to current state of the art deflated SAP solvers used in lattice QCD [69, 70]. Finally, we test whether the FETI algorithms serves as a suitable preconditioner to other solution strategies.

The effectiveness of different algorithms and setups can be assessed in a number of different ways. Most apparently the execution time is a suitable measure to compare different setups. However, the execution time depends on a number of factors apart from the algorithm itself. The computer architecture, parallelization and implementation of the algorithm influence the execution time. For that reason it is beneficial to also consider the number of iterations an algorithm takes to reach a certain precision.

When analyzing the FETI algorithm in the lattice QCD application we will consider the following aspects. In section 10.1 we start by describing the Dirac system that we want to solve. We introduce the gauge ensemble and make some general remarks on the FETI system and various subsystems. Then in the first part of the analysis we consider the system of the Lagrange multipliers. This is the global FETI system given by eq. (9.52). In section 10.2.1 we employ the FETI algorithm as a solver and analyze its convergence behavior. Next, in section 10.2.2, we take inspiration from [144] and use the FETI algorithm as a preconditioner to a global iterative solver. We define preconditioners by decreasing the precision of the FETI algorithm as well as by taking certain approximations inside the FETI algorithm itself. At this point we compare the algorithm with current state-of-the-art solvers used in lattice QCD applications [69, 70] Finally, in section 10.3, we have a close look at the FETI block system. We analyze its condition and convergence before reaching a conclusion where we compare this work to other solver algorithms

L	T	β	κ_l	κ_s
32	64	3.46	0.1369814	0.136408545

Table 10.1: Parameters of the CLS [78] gauge ensemble `B451`. We give the spacial and temporal extent, the coupling β as well as the κ parameters for the light and strange quarks.

in lattice QCD and other applications of the FETI algorithm.

10.1 Setup

The relevant system of equations in lattice QCD is given by the Wilson Dirac operator defined in eq. (1.30). The Wilson Dirac operator, later only called Dirac operator, includes only next neighbor interactions. We can separate the diagonal part of the Dirac operator in the so-called hopping expansion.

$$D = C(\mathbb{1} - \kappa H) \tag{10.2}$$

The component C is diagonal in the space-time coordinates, depends on the quark mass and the improvement term defined in section 1.2. It contains the diagonal part of the operator. The hopping matrix

$$H = \sum_{\mu} (1 - \gamma_\mu)\, U_\mu(x)\delta_{y,x+a\hat{\mu}} + (1 + \gamma_\mu)\, U_\mu^\dagger(x - a\hat{\mu})\delta_{y,x-a\hat{\mu}} \tag{10.3}$$

contains the next neighbor interactions and the gauge links $U_\mu(x)$ attached to the point x in the μ direction. The hopping κ parameter

$$\kappa = \frac{1}{2am_0 + 8} \tag{10.4}$$

depends on the bare quark mass and influences the condition of the Dirac system.

The gauge links $U_\mu(x)$ are taken from an ensemble generated by the CLS group [78]. The internal ensemble ID is `B451r000`. The lattice has $64 \times 32^3 = 2097152$ sites and uses periodic boundary conditions in time and space. The fermion fields are subject to antiperiodic boundary conditions in time. The parameters of the lattice are listed in table 10.1. The coupling $\beta = 3.46$ corresponds to a lattice spacing of $a \approx 0.07\,\mathrm{fm}$ as calculated in chapter 5 and [1]. The parameters κ_l and κ_s correspond to a Pion mass of $m_\pi = 418\,\mathrm{MeV}$ and a Kaon mass of $m_K = 572\,\mathrm{MeV}$ as calculated in [164]. In the following we will analyze the Dirac operator for the light quark. Since we are interested in algorithmic quantities, we do not average over several gauge configurations. All the measurements are taken on one gauge configuration (`n1000`) of this ensemble. While this prohibits taking any gauge averages key properties of the solver algorithms can still be observed.

10.1.1 Problem Sizes

Let us now compare the FETI-system in eq. (9.45) to the original system from eq. (9.29). The volume of the global lattice is $V = \prod_i^4 N_i$. The resulting linear system has dimension $12V$ since a spinor with 12 components resides on each lattice site. The size of the FETI-system depends on the size of the blocks used. The number of lattice points in each direction of the FETI block is called s_b^i. Because the FETI blocks overlap by one lattice spacing they are spaced $s_b - 1$ lattice

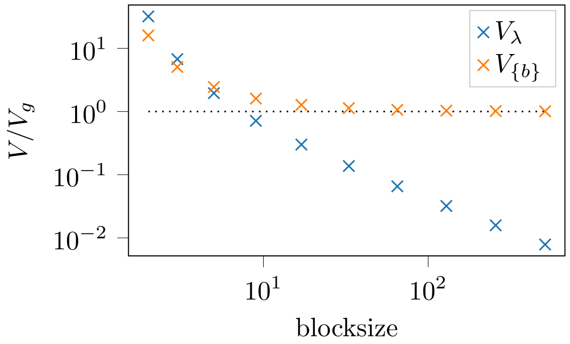


Figure 10.1: Different volumes normalized by the size of the original system for various block sizes. The size of the λ-system V_λ is shown in blue and decreases for increasing block size. The λ-system is only smaller than the original system for block sizes greater than 4. The total number of lattice points on all blocks $V_{\{b\}}$ normalized by the global volume is shown in orange.

spacings apart, as shown in fig. 9.2. It is always assumed that the blocks evenly divide the global lattice. The volume of a FETI block is

$$V_b^{\text{FETI}} = \prod_i^4 s_b^i. \tag{10.5}$$

The block boundary in direction i has the volume $\prod_{j\neq i}^4 s_j = V_b/s_i$. Including the boundary in up and down direction the total number of points on the block boundary is

$$V_\Delta^b = 2\sum_i \frac{V_b}{s_b^i}. \tag{10.6}$$

The number of blocks in direction i is $n_b^i = N_i/(s_b^i - 1)$. Accordingly, the total number of blocks is

$$n_b = \prod_i^4 n_b^i = \prod_i^4 \frac{N_i}{s_b^i - 1}. \tag{10.7}$$

Each Lagrange multiplier connects points on two block boundaries. The total number of Lagrange multipliers needed is therefore

$$\begin{aligned} V_\lambda &= \frac{1}{2} n_b V_\Delta^b \\ &= \left(\prod_i^4 \frac{N_i s_b^i}{s_b^i - 1}\right)\left(\sum_i^4 \frac{1}{s_b^i}\right). \end{aligned} \tag{10.8}$$

The relative system size is independent of the size of the original system and only depends on the block size s_b^i.

$$\frac{V_\lambda}{V} = \left(\prod_i^4 \frac{s_b^i}{s_b^i - 1}\right)\left(\sum_i^4 \frac{1}{s_b^i}\right) \tag{10.9}$$

This quantity is shown in fig. 10.1 in blue. We can see that the system size is increased if the blocks are chosen too small. Only for block sizes of $s_b^i > 5$ the FETI system is smaller than the original system.

s_b	N_b	V_b	V_λ
2	2097152	16	67108864
3	131072	81	14155776
5	8192	625	4096000
9	512	6561	1492992
17	32	83521	628864
33	2	1185921	287496

Table 10.2: Algorithmic quantities for the FETI algorithm for different block sizes s_b. We list the number of blocks N_b, the block volume V_b and the size of the λ system V_λ. The volume of the global lattice is $V = 64 \times 32^3 = 2097152$. Note that the dimension of the respective linear system is $12V$, since a spinor with 12 components is attributed to each point.

Similarly, we consider number of points added by duplicating the boundaries. The ratio

$$\frac{V_{\{b\}}}{V_g} = \frac{n_b V_b}{V_g} = \prod_i^4 \frac{s_b^i}{s_b^i - 1} \tag{10.10}$$

compares the volume of all blocks to the global volume. It is shown in orange in fig. 10.1. We see that the ratio approaches 1 as the blocks get bigger. That means that the overall increase of the problem due to the duplication of the boundary points becomes less relevant for larger block sizes.

This first analysis shows that we can only hope to improve the original system if

(a) the FETI system is significantly better conditioned than the original system

(b) the FETI blocks are chosen large enough such that the workload is shifted from the global λ system to the block systems which can be solved in parallel.

The conditioning of the FETI system (a) will be discussed in section 10.2.1. The condition (b) leads to a balancing problem. On one hand, the blocks need to be large enough for the algorithm to be efficient. On the other hand, if the blocks are too large, one runs into the difficulty of efficiently finding the block solutions. Additionally, if the blocks are very large, there is fewer of them, which makes it harder to treat them in parallel.

Table 10.2 lists the number of FETI blocks, the block volume and the size of the FETI system for various block sizes. We can see that only for block sizes greater than 5 does the FETI system become smaller than the original system. For a block size of $s_b = 33$ there are only two blocks. The parallel efficiency is decreased dramatically. In the following we consider block sizes of $s_b = 9$ and $s_b = 17$.

10.1.2 Solvers for the FETI Algorithm

The FETI algorithm requires the solution of linear systems at various stages. On the outermost level the global system of Lagrange multipliers λ introduced in eq. (9.52) must be solved. This system is also called FETI-system or λ-system. The global FETI system typically needs to be solved to high precision if the algorithm is used as a direct solver. The convergence of the λ system if the FETI algorithm is used as a solver is analyzed in section 10.2.1. When the FETI algorithm is used as a preconditioner the precision of the solution can be drastically lowered. This case is discussed in section 10.2.2.

Inside the $F_{\lambda\lambda}$ operator, the block Dirac operator needs to be inverted (see eq. (9.53)). Clearly the global FETI system can not be solved to a higher precision than that used in the inversion of the block operator inside $F_{\lambda\lambda}$. Because of the nested inversions, the block system is solved for each step of the outer solver. For that reason, any improvement to the block system immediately speeds up the overall calculation. Section 10.3 is dedicated to analyzing different solvers for the block operator. On the same level the block operator needs to be solved for the preparation of the right-hand side $\tilde{f}$ in eq. (9.54) and the resubstitution in eq. (9.48). Since it is unclear how the precision of the solution of the FETI system translates to the precision of the original problem, it can be beneficial to solve these two steps to a slightly higher precision than the one used for the $F_{\lambda\lambda}$ operator and the global inversion.

Finally, the Dirichlet preconditioner requires an additional solution of the bulk-to-bulk part of the block operator in the Schur complement (see eqs. (9.59) and (9.61)). Since this inversion is only used for preconditioning, a much lower precision is required.

10.2 The λ System

In this section we will consider the solution of the FETI system in eq. (9.52). Ultimately we are interested in the convergence properties of the FETI system. Therefore, at this stage, we will take the inversion of the block system for granted. It will be discussed in detail in section 10.3. Once the system is solved, we know the value of the Lagrange multipliers. From those we can obtain the solution of the original system by resubstituting using eq. (9.48).

In section 10.2.1 we will employ the FETI algorithm as a direct solver and compare it to the global GCR and the SAP implemented in the `openQCD` package [123]. We will analyze the convergence of the different algorithms and setups. Section 10.2.2 is dedicated to analyzing how we can use the FETI algorithm as a preconditioner to the global solver.

10.2.1 FETI as Direct Solver

In this section we will use the FETI algorithm as a direct solver. We draw a random right-hand side to the Dirac equation

$$D\psi = \eta \tag{10.11}$$

and prepare the right-hand side for the FETI system $\tilde{f}$ according to eq. (9.54). The Dirac operator uses the light quark mass, i.e. $\kappa = \kappa_l$. After the preparation of the source we invert the $F_{\lambda\lambda}$ operator using the GCR algorithm. We try versions without preconditioning and using the Lumped and Dirichlet preconditioners defined in section 9.2.6. Having found the appropriate Lagrange multipliers λ we obtain the solution to the original problem by resubstituting using eq. (9.48). As a reference we also invert the same system using the global GCR algorithm. Figure 10.2 shows the relative residue as a function of the iteration. In the accompanying Table 10.3 we present the slopes of the lines shown in fig. 10.2. The column $\frac{n_{\text{iter}}}{\text{dec}}$ lists the number of iterations per decade, i.e. the number of iterations needed to lower the residue by a factor of 10. The measurements of the slope are taken once the convergence reaches a linear (on the log scale) behavior.[1]

The reference inversion using the global GCR algorithm is shown in red in fig. 10.2. We can see that in the beginning the residue falls off quickly. For higher iterations the progress is much slower. This is reflected in the high iteration counts in table 10.3. The regular FETI algorithm suffers from the same problem for both block sizes. Additionally, the regular FETI algorithm

[1] The global GCR algorithm never reaches this linear shape. Here the measurement is taken at $n_{\text{iter}} = 200$ and understood as a lower bound.

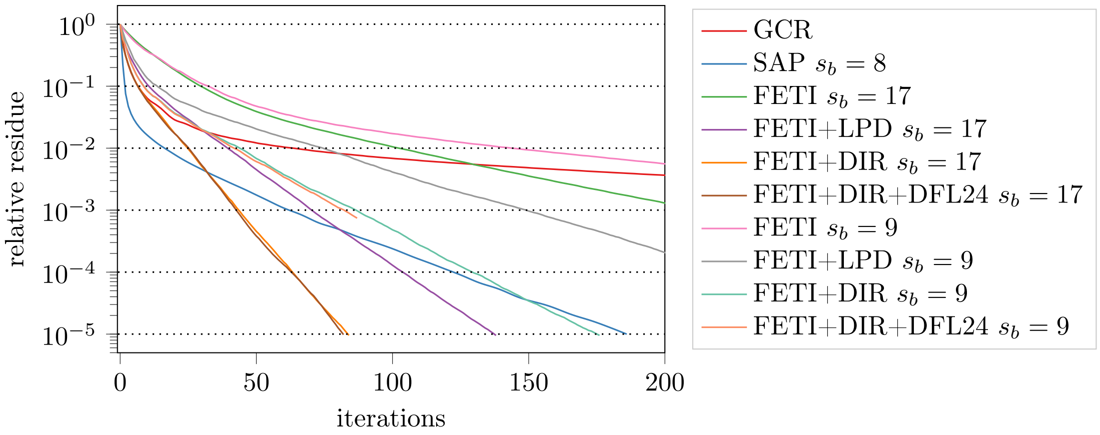


Figure 10.2: Relative residue as a function of the iteration for various solvers, preconditioners and block sizes. Measurements taken at $\kappa = \kappa_l$.

	Algorithm	block size	$\frac{n_{\text{iter}}}{\text{dec}}$	$\frac{\text{time}}{\text{dec}}[s]$
—	GCR	-	> 431	> 35
—	SAP	8	64.7	255.1
—	FETI	17	117.4	4292.5
—	FETI+LPD	17	34.1	2792.7
—	FETI+DIR	17	20.4	1650.2
—	FETI+DIR+DFL24	17	20.5	4013.5
—	FETI	9	233.0	5648.9
—	FETI+LPD	9	75.5	3556.6
—	FETI+DIR	9	48.1	2136.9
—	FETI+DIR+DFL24	9	41.1	8229.3

Table 10.3: Overview of Algorithm run times and iteration counts for $\kappa = \kappa_l$. We list the number of iterations and runtime per decade, i.e. per 10^{-1} decrease of the residue. The slope of the lines in fig. 10.2 directly correspond to column 3.

exhibits slower convergence in the beginning and only surpasses the global GCR late. Only the addition of the preconditioners ameliorates this problem. The Lumped preconditioner is cheaper than the Dirichlet preconditioner, but the convergence is faster for the Dirichlet. In the end the Dirichlet preconditioner leads to a faster convergence than the Lumped preconditioner as can be seen from table 10.3. The addition of the deflation preconditioner with 24 deflation vectors yields next to no improvement over the Dirichlet preconditioner alone. The inclusion of the deflation preconditioner, however, adds computational intensity to the algorithm. It is for that reason that, in the following, we refrain from using the deflation preconditioner for the FETI algorithm.

For all preconditioners the larger blocks with block size 17 work better than block with size 9. This is in line with the analysis in section 10.1.1 where we compared the sizes of the FETI system for different block sizes.

The last column in table 10.3 lists the time needed to decrease the residue by a factor of 10. The execution times are of course subjective to the hardware the tests are performed on, the level of parallelization and the implementation. They are not meant as absolute measurements, but only as guidelines to compare the different algorithms roughly. Still we can see that the execution times for the FETI algorithms pose a serious problem. The run times for the FETI algorithm are upwards of 22 times larger than for the global GCR. This outweighs all the savings made by the lower iteration counts. These high execution times stem from the nested inversions. In eq. (9.53) we can see that for each application of the $F_{\lambda\lambda}$ operator, i.e. in every iteration of the outer solver, the set of block operators needs to be inverted. This leads to a considerable computational workload in each individual step of the inversion of the $F_{\lambda\lambda}$ operator. If there is no way to speed up the solution of the block system, we can not hope to use the FETI algorithm as a competitive solver. The block system will be analyzed in section 10.3.

The SAP algorithm suffers from a similar complication. Here, when employing the SAP as a solver, we solve the block system with 100 iterations of the MRES algorithm. The residue for each cycle of the SAP is shown in fig. 10.2. In the last column of table 10.3 we can see that the execution time for the SAP algorithm is about a factor 7 larger than for the global GCR. In the first few iterations of the SAP algorithm the residue is reduced dramatically. Because of this low precision behavior the SAP algorithm is used as a preconditioner to the global GCR algorithm in [143, 144]. In the following section we will analyze how and in what way the FETI algorithm can be salvaged to define a preconditioner for the global system.

10.2.2 FETI as Preconditioner

As we have seen in the previous section, neither the FETI algorithm nor the SAP algorithm are suited as a direct solver in this application. While the number of iterations is low, the execution time is exceedingly long. This is due to the very expensive application of the $F_{\lambda\lambda}$ operator and the inversion of the block system within. In this section we follow the approach by Lüscher [70, 144] and utilize the domain decomposition solver algorithm as a preconditioner to the global solver. First we use the FETI algorithm to a very low precision to precondition the global GCR solver. In fig. 10.2 we can see that the convergence in the beginning is very steep. The FETI algorithm is efficient only in this regime. In a second approach we build a preconditioner to the global GCR algorithm out of the preconditioners for the FETI algorithm introduced in section 9.2.6. We have seen the effect of these preconditioners on the FETI algorithm in the previous section (see table 10.3). Here, we analyze if and how we can adapt these preconditioners to work for the global system.

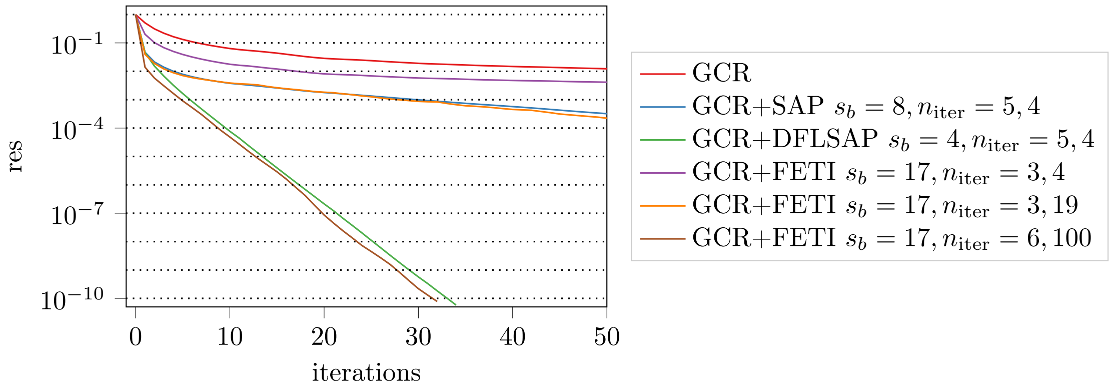


Figure 10.3: Convergence of the global GCR with various preconditioners. We specify the number of iterations as $n_{\text{iter}} = n_{\text{iter}}^{\text{outer}}, n_{\text{iter}}^{\text{block}}$. The measurements are done for $\kappa = \kappa_l$.

Low Precision FETI

Taking inspiration from the way the SAP is used as a preconditioner to the GCR algorithm in [144], we use the FETI algorithm to a very low precision as a preconditioner to the global GCR solver. Rather than specifying the relative precision of the solver, we truncate the solver by specifying the number of iterations. We set the number of iterations of the outer solver (SAP or FETI) as well as the block solver. Typically, the SAP preconditioner is used with 5 outer iterations and 4 iterations for the block solver. If we want to use the same number of iterations for the FETI algorithm, we are left with 3 iterations for the outer solver, since there is one inversion of the block system in each of the preparation and resubstitution steps. Figure 10.2 gives a sense of the precision reached after the respective number of iterations.

For the SAP algorithm it is advantageous to use smaller blocks, because the block system converges faster. In section 10.2.1 we have seen that the FETI algorithm prefers larger blocks. For that reason we compare the different algorithms with block sizes that are best for the individual algorithm. In this case we use $s_b^{\text{FETI}} = 17$ for the FETI preconditioner and $s_b^{\text{SAP}} = 8$ for the SAP preconditioner.

In sections 8.3 and 9.1.1 we presented deflation techniques and the deflated SAP (DFLSAP) solver that is implemented in `openQCD` [69, 70, 123]. Similar to the FETI and SAP algorithms we use it here as a preconditioner to the global GCR solver. We are using $N_s = 20$ deflation modes to span the deflation subspace. These modes are calculated on blocks with side length $s_b^{\text{dfl}} = 4$. The SAP algorithm that is used for the remaining deflated system uses the SAP block size $s_b^{\text{SAP}} = 8$ described above.

Figure 10.3 shows the convergence for different preconditioners to the global GCR algorithm. In table 10.4 we give the number of iterations and execution time per decade of the relative residue. These values correspond to the slope of the lines in fig. 10.3 and are used to compare the preconditioners. The inversions were executed for the light quark mass. As a baseline we give the convergence behavior for the global GCR algorithm, the SAP preconditioner with 5 outer and 4 block iterations and the deflated SAP (DFLSAP) preconditioner with 5 outer and 4 block iterations. The deflation subspace is spanned by 24 vectors calculated from repeated application of the SAP preconditioner and orthogonalization. The SAP preconditioner is applied 10 times to suppress higher modes. Details on the procedure are supplied in [69].

Comparing with the previous section we note that using the low precision FETI algorithm as a preconditioner to the global GCR solver leads to an improved convergence in all cases. This is

	Algorithm	block size	n_{iter}		$\frac{n^{\text{g}}_{\text{iter}}}{\text{dec}}$	$\frac{\text{time}}{\text{dec}}[s]$
			global	block		
—	GCR	-	-	-	> 124	> 10
—	GCR+SAP	8	5	4	40.1	35.9
—	GCR+DFLSAP	4	5	4	3.9	4.7
—	GCR+FETI	17	3	4	164.7	650.7
—	GCR+FETI	17	3	19	38.5	569.5
—	GCR+FETI	17	6	100	4.0	347.9
—	GCR+LPD	17	-	-	36.6	190.4
—	GCR+DIR	17	-	20	32.5	168.0
—	GCR+DIR	17	-	40	18.6	160.9
—	GCR+DIR	17	-	60	15.1	183.5

Table 10.4: Overview of convergence results for different preconditioners to the global GCR algorithm. We list the number of iterations and runtime per decade, i.e. per 10^{-1} decrease of the residue. The colors correspond to the lines in figs. 10.3 and 10.5.

true for the number of iterations as well as for the execution time. The convergence is improved because in every step of the outer GCR solver we benefit from the rapid convergence of the first few FETI iterations (see fig. 10.2).

We now compare different preconditioners to the global GCR solver. The SAP preconditioner uses 5 outer and 4 inner iterations. Because of the two inversions in the preparation and resubstitution in the FETI algorithm, this corresponds to the FETI algorithm with 3 outer and 4 inner iterations. This setup performs significantly worse than the SAP preconditioner. The convergence behavior is similar to the global GCR algorithm. To get a convergence similar to the SAP preconditioner we have to increase the block iterations to 19 for the FETI algorithm. This means increasing the overall number of applications of the block operator by a factor of 4.75. This behavior is due to the increased condition number of the block system (see sections 9.2.4 and 10.3). To match the deflated SAP preconditioner (DFLSAP) we have to increase the number applications of the block operator by a factor of 40. Clearly without a way to efficiently invert the block system, the low precision FETI algorithm is not an effective preconditioner.

FETI Preconditioners

In this section we construct a preconditioner to the global GCR using the FETI geometry and structure. The three steps of the FETI algorithm presented in section 9.2 are the preparation of the right-hand side $\tilde{f}$, the solution of the FETI system $F_{\lambda\lambda}$ and the resubstitution to obtain the solution to the original problem. Here we replace the expensive computation of $F_{\lambda\lambda}^{-1}$ by the application of the preconditioners $F_{\lambda\lambda}^{\text{LPD}}$ and $F_{\lambda\lambda}^{\text{DIR}}$ presented in section 9.2.6. To get a sense of how well these preconditioners approximate the inverse $F_{\lambda\lambda}^{-1}$ we calculate the correlator

$$C(t) = |M\psi(t_0)|^2\,(t) \tag{10.12}$$

where $\psi(t_0)$ is a point source on the time slice t_0. The operator M is set to various preconditioners. The correlator $C(t)$ is shown in fig. 10.4. The source $\psi(t_0)$ is positioned at $t_0 = 0$. As a reference in red we show the correlator with $M = D^{-1}$. Next we set M to the different

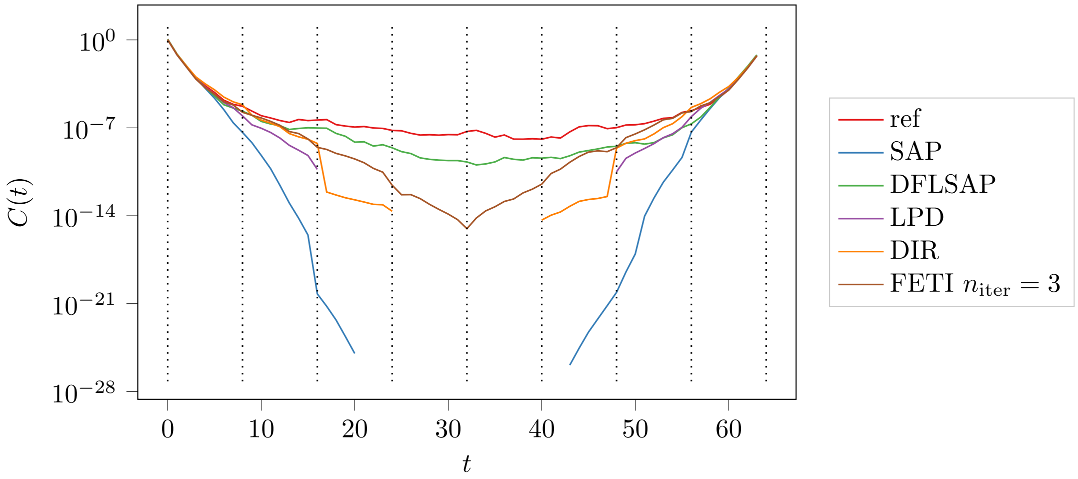


Figure 10.4: Correlator $C(t)$ for various preconditioners. The source is located at $t_0 = 0$. The red line labeled 'ref' is the reference correlator with $M = D^{-1}$. The FETI block boundaries are indicated as vertical lines. We can see in which regions of the lattice the individual preconditioners perform well.

preconditioners and compare how closely they match the reference line. In order to compare the SAP and FETI-type preconditioners, we use a block size of 8 and 9 respectively. For the SAP preconditioner defined in eq. (9.9) a steep drop already inside the block that the source is located can be seen. This is due to the low precision that the block system is solved for the SAP preconditioner. Using 4 outer iterations for the SAP algorithm means that the boundaries are only communicated back and forth 4 times and the contribution from the source can only reach a distance of 3 blocks. For any points a distance of 3 blocks away from the source, the correlator is $C(t) = 0$. The Lumped preconditioner communicates the boundaries once in the preparation of the source $\tilde{f}$ and once in the resubstitution. Accordingly, the contribution only reaches 2 blocks. The Dirichlet preconditioner reaches one block further because of the additional inversion of the internal-to-internal block operator. We also show the full FETI algorithm with the same number of outer iterations as the SAP and DFLSAP algorithms. This setup was used in the previous section. It can overcome the gap in the center of the lattice and benefits greatly from the more precise block solutions. However, as discussed before, the full FETI algorithm is prohibitively expensive. Lastly we show the deflated SAP algorithm. With only a very small number of 5 outer iterations and rather inaccurate block solutions (4 iterations of the MRES) it matches the reference correlators most precisely. The deflation subspace is spanned by 10 vectors that are computed beforehand. As presented in sections 9.1.1 and 9.1.2 these vectors consist of the approximate low modes of the Dirac operator. They are calculated using distillation, i.e. the repeated application of the SAP preconditioner and subsequent orthogonalization. Applying the SAP preconditioner with 5 cycles and 4 inner iterations 5 times sufficiently suppresses the higher modes of the Dirac operator.

The correlator only gives an inaccurate picture of the preconditioners, as the computational cost for the different preconditioners is not considered. For this reason we will also analyze the convergence behavior of the preconditioners made from the FETI algorithm. Figure 10.5 shows the convergence of the global GCR algorithm with the FETI type preconditioners. We can see that in terms of iterations the FETI type preconditioners can not significantly improve over the SAP preconditioner. Table 10.4 lists the slopes of the convergence for various solvers and precon-

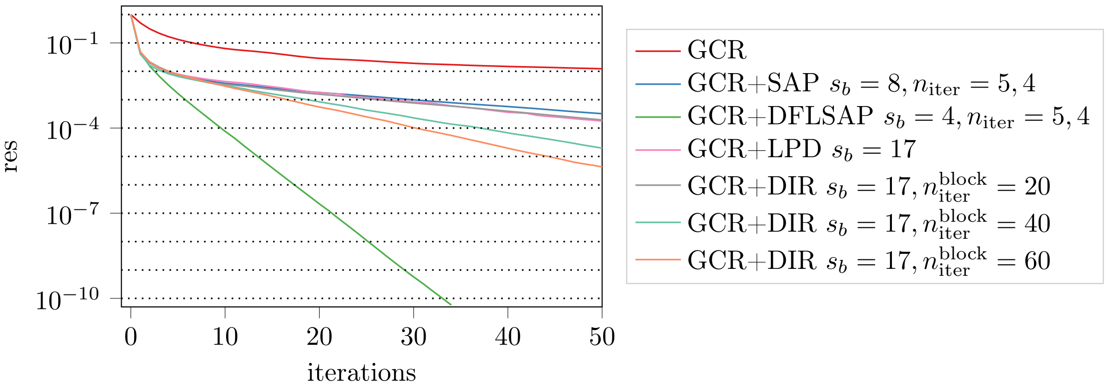


Figure 10.5: Convergence of the global GCR with various preconditioners. We specify the number of iterations as $n_{\text{iter}} = n_{\text{iter}}^{\text{global}}, n_{\text{iter}}^{\text{block}}$. The measurements are done for $\kappa = \kappa_l$.

ditioners. We can see that the preconditioners presented in this section are an improvement over using the full FETI algorithm as a preconditioner. The Lumped and Dirichlet preconditioners are significantly cheaper to compute compared to the full inversion of the $F_{\lambda\lambda}$ operator. For that reason the time to decrease the residue by a factor of 10 is smaller compared to the full FETI preconditioner even if the number of iterations is higher. However, already the preparation and resubstitution steps in the FETI algorithm are more computationally expensive than the application of the SAP and DFLSAP preconditioners. Similar to the previous section this is due to the slow convergence of the transformed FETI block system. For the FETI algorithm to be a viable alternative to the SAP and DFLSAP algorithms we must control the block system.

10.3 The Block System

In this section we will analyze the block system. It plays a central role in the FETI algorithm. On one hand we want to shift the main workload to the block system, because the blocks can be solved independently of each other. On the other hand the block system directly affects the overall performance of the FETI algorithm. Since it appears in the definition of the global operator $F_{\lambda\lambda}$ in eq. (9.53) it needs to be solved for every application of the operator. We already encountered difficulties with the performance of the block system in sections 10.2.1 and 10.2.2 where we concluded that many of the problems with the FETI algorithm originate from the slow convergence of the block system. It is therefore worthwhile to closely study the block system.

10.3.1 Explicit Solvers

The block solver needs to solve relatively small problems. It needs to be as fast as possible, while retaining stability. An ideal match for the requirements to the block solver are explicit solvers. They are inherently stable and once the inverse matrix is calculated, its application is vastly more efficient than iterative solvers. However, the memory required to store the inverse operators is prohibitively large for four dimensional applications. Table 10.5 gives an overview of the memory requirements for different block sizes. We assume that the operators are stored in single precision. Small blocks use less total memory, since the block size enters to the fourth power. Using the smallest block size of $s_b = 2$, which lead to a very inefficient FETI algorithm (see section 10.1.1),

		Memory	
s_b	N_b	one block	all blocks
2	2097152	576.0 Kb	1.1 Tb
3	131072	14.4 Mb	1.8 Tb
5	8192	858.3 Mb	6.7 Tb
9	512	92.4 Gb	46.2 Tb
17	32	14.6 Tb	467.8 Tb
33	2	2.9 Pb	5.8 Pb

Table 10.5: Memory requirements for storing the inverted block operators in single precision. This example assumes a 64×32^3 global lattice.

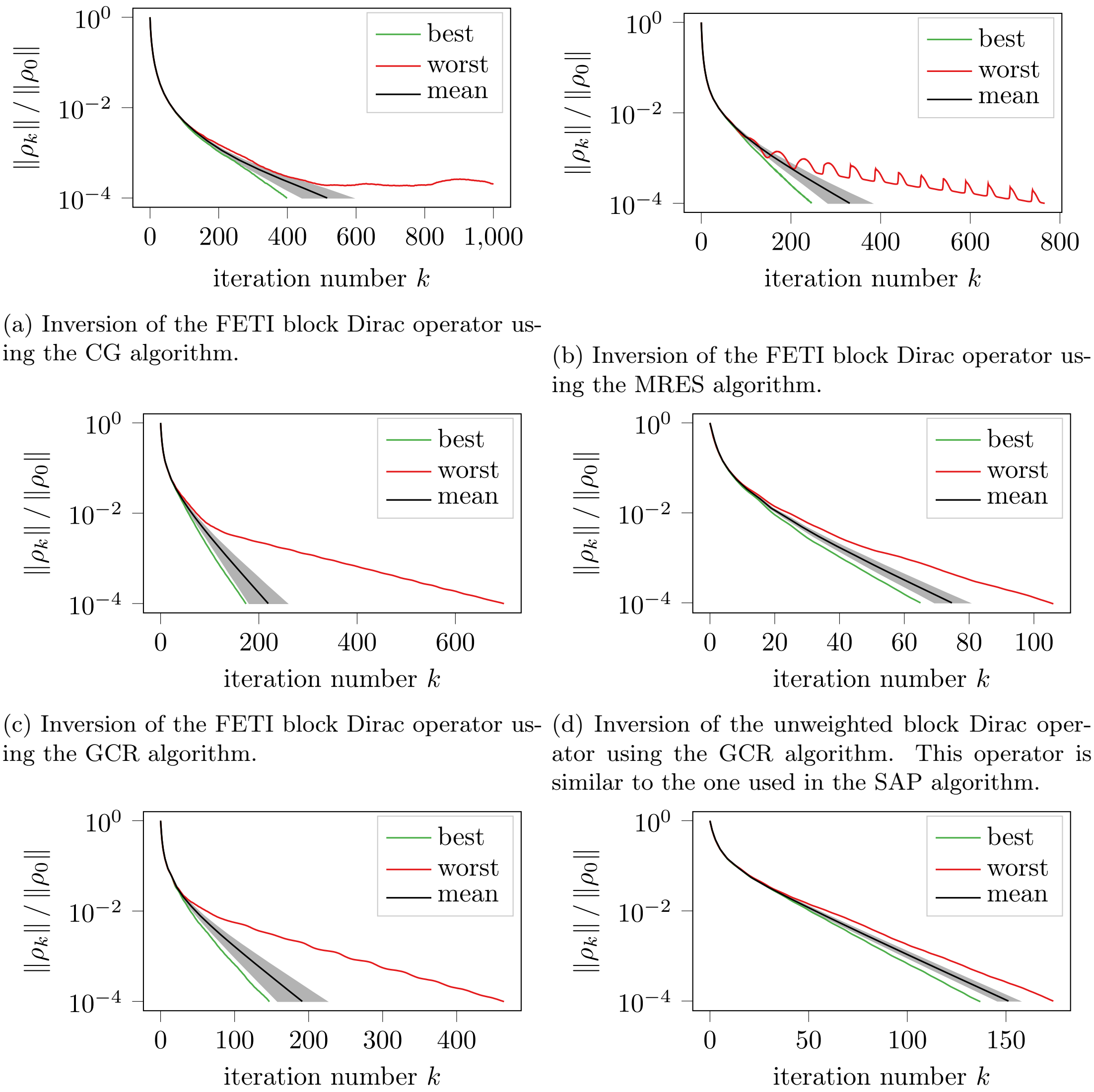
solver	system	$\frac{n_{\text{iter}}}{\text{dec}}$	$\frac{\text{time}}{\text{dec}}\,[s]$
CG	FETI blocks	291.42	2.21
MRES	FETI blocks	165.91	0.65
GCR	FETI blocks	80.92	0.49
GCR	SAP blocks	28.36	0.17
GCR	FETI blocks preconditioned with W_Δ	75.66	0.46
GCR	FETI blocks preconditioned with 1 deflation vector	75.24	0.74
GCR	FETI blocks preconditioned with 4 deflation vectors	67.10	1.72
GCR	FETI blocks preconditioned with 24 deflation vectors	49.66	14.94

Table 10.6: Slopes of the convergence of various block systems and solvers. Out of the solvers for the FETI blocks, the GCR performs best. Preconditioning the GCR is able to decrease the number of iterations but not the execution time. This is in part due to implementation.

still requires 1.1 Terabytes of memory to store the operators for all blocks. With bigger blocks that have proven beneficial for the global FETI system the memory requirements rise to levels that are currently unrealistic. With explicit solvers out of the question we have to resort to iterative methods again.

10.3.2 Comparing Iterative Solvers

In this section we compare the CG, GCR and MRES solvers presented in section 8.1. In figs. 10.6a to 10.6c we see the residue of the solution plotted as a function of the number of steps for the three solvers. The Dirac operator uses the mass of the light quark. This is the most difficult of the cases we consider here. Below, we analyze how the situation changes if heavier quarks are used. We consider blocks of size $V_b = 8^4$. Accordingly, there is a total of 512 blocks on the lattice. Out of these the one that reaches the desired precision in the fewest iterations is plotted in green. The block that converges the slowest is shown in red. The average over all blocks is shown in black. The slopes of the average lines are given in table 10.6 in units of number of iterations and execution time per decade. From the first three rows we see that the GCR algorithm has the lowest average number of iterations. It is worthwhile to not only consider the average behavior of the solvers. In figs. 10.6a to 10.6c we see that up to a relative tolerance of 10^{-2} all solvers perform well and do not stray far from the average. Going beyond there are differences between the solvers. The MRES and CG solvers develop instabilities, where the residue does not decrease monotonically. This behavior can be caused by insufficient numerical



(a) Inversion of the FETI block Dirac operator using the CG algorithm.

(b) Inversion of the FETI block Dirac operator using the MRES algorithm.

(c) Inversion of the FETI block Dirac operator using the GCR algorithm.

(d) Inversion of the unweighted block Dirac operator using the GCR algorithm. This operator is similar to the one used in the SAP algorithm.

(e) Inversion of the FETI block Dirac operator using the GCR algorithm and the inverse of the weight matrix W_Δ as an explicit preconditioner.

(f) Inversion of the FETI block Dirac operator using the GCR algorithm and the deflation preconditioner presented in section 10.3.6. Here 24 deflation modes are used.

Figure 10.6: Comparison between different solvers for the weighted block Dirac operator at $\kappa = \kappa_l$. Average values are taken over all blocks on the lattice (512). Figures (a), (b) and (c) show the convergence of the FETI block Dirac operator using the algorithms presented in section 8.1. In figure (d) we show the convergence of the block Dirac operator without the weights introduced in section 9.2.4. The solver used is the GCR. This Dirac operator without the weights is the one used in the SAP algorithm. Figure (e) depicts the convergence of the FETI block Dirac operator, i.e. with the weights. Here we use the GCR algorithm that is preconditioned with the diagonal weight matrix introduced at the end of section 9.2.1.

precision as the block solvers are only executed using single precision floating point arithmetic. In practice the instabilities of the MRES solver are not critical it is only used with a very limited number of iterations (< 100) inside the SAP. Only the GCR solver is able to reach the desired accuracy for all blocks. The GCR solver also has the lowest average number of iterations among the three. However, the spread between the best and worst solutions is large. Since we require all block solutions, the overall performance is impacted by these slow blocks. The average number of iterations is close to the lowest. This suggests that the majority of the blocks are well-behaved with only a small number of badly conditioned blocks. We will analyze this theory in more depth in section 10.3.5.

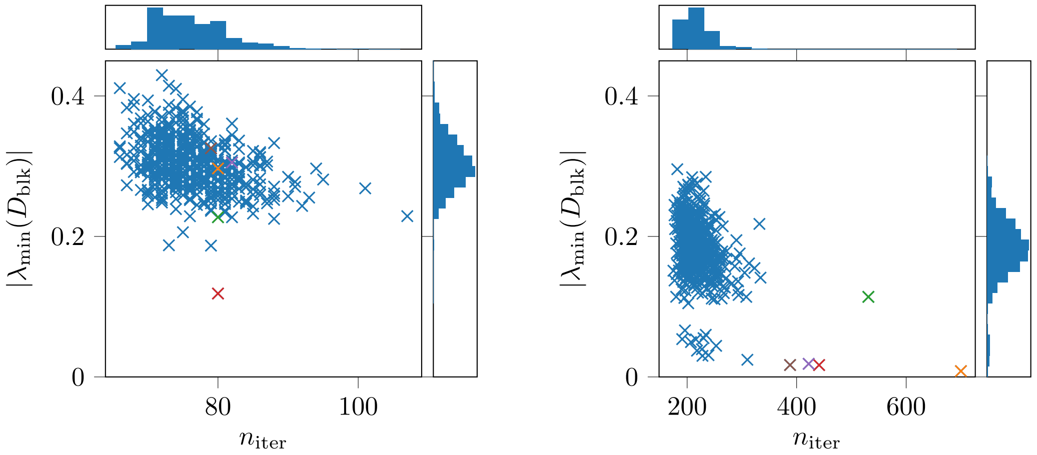
10.3.3 Weights in the Dirac Operator

In section 9.2.4 we introduced the weights in the block Dirac operator. They are needed for the block system to be equivalent to the original global system. These weights consist of factors $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}$ and $\frac{1}{16}$ in the boundary sites. These factors increase the condition number of the block operators. In this section we analyze the impact of these weights on the solver performance.

In fig. 10.6 we show the residue of the solution as a function of the number of iterations. As in the previous section we consider 512 blocks of size $V_b = 8^4$. We consider the GCR solver for the block system as it was deemed most suitable in the previous section. We show the curves for the fastest and slowest converging blocks as well as the average over all blocks. In fig. 10.6d we show the regular block Dirac operator as a baseline. Here we set $w = 1$, effectively disregarding the weights introduced in section 9.2.4. The corresponding line in table 10.6 is labeled with SAP blocks. The number of iterations per decade of the residue drops significantly. In fig. 10.6c and row three of table 10.6 we show the convergence for the weighted block Dirac operator as it is used in the FETI algorithm. The weights are $w = 1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}$ as described in section 9.2.4. The introduction of the weights increased the number of iterations for the best and average convergence lines in roughly the same way. The number of iterations is increased roughly threefold compared with the SAP Dirac operator. This in itself is a problem that is very hard to overcome. The convergence of the FETI system would have to be three times faster just to break even with the increased workload on the blocks. On top of that the introduction of the weights exacerbates the already slow convergence of the most ill-conditioned blocks. While the average number of iterations is increased by a factor of 3, the number of iterations for the slowest blocks (red line) is increased by more than a factor of 6.

10.3.4 Diagonal Preconditioners

We now try to find a solution to the slower convergence of the block system caused by the weights in the FETI block Dirac operator. In section 8.2 we discussed explicit preconditioners. They can be used to negate a part of the effect of the weights on the operator. In section 9.2.4 we introduced weights on the sites and the links connecting neighboring sites. Here we will focus on the weights of the sites. They are conceptually easier because they only modify the diagonal of the Dirac operator. Additionally, they go down to $\frac{1}{16}$ whereas the smallest link weights are $\frac{1}{8}$. The diagonal matrix W_Δ contains the factors that scale the diagonal entries of the FETI block Dirac operator. We can use the inverse W_Δ^{-1} as a preconditioner. This amounts to scaling all columns of the weighted Dirac operator such that its diagonal entries coincide with those of the original operator. This works very well if large quark masses are chosen. In this case the Dirac operator is dominated by its diagonal. For more realistic quark masses the operator is no longer dominated by its diagonal. Therefore, this preconditioning technique is not as effective. In fig. 10.6e we show the convergence behavior of the preconditioned system. Comparing with



(a) No weights in block Dirac operator D_{blk}.

(b) Weighted block Dirac operator D_{blk}.

Figure 10.7: Comparison of smallest block Eigenvalues and iteration counts for the GCR algorithm. The block size is $s_b = 8^4$, which results in 512 blocks. On the left, in figure (a), we see the smallest eigenvalues of the block Dirac operator without the weights introduced in the FETI algorithm. On the right, in figure (b), the weights are included in the block Dirac operator. The colored points indicate the blocks with the largest iteration count for the block Dirac operator that includes the weights.

fig. 10.6c, where we show the setup without the diagonal preconditioner, we see that there is very little to be gained for the average number of iterations. Only a very small number of iterations can be saved on average. This can also be seen from rows three and five in table 10.6. However the preconditioner somewhat improves the slow convergence of the most ill-conditioned blocks. The spread between the fastest and slowest convergence is halved compared to the solver with no preconditioning in fig. 10.6c.

In conclusion the diagonal preconditioner is not sophisticated enough to capture the structure of the weights in the operator and ameliorate their effects.

10.3.5 Conditioning of the Block System

We have seen that the block system plays a critical role in the performance of the FETI algorithm. In this section we analyze the block system not by its convergence properties as in the previous sections but by its spectrum and condition. The condition of the block system is mediated by the smallest eigenvalues λ_{min} of the block operator. The absolute value of the smallest eigenvalue of the block Dirac operator is shown in fig. 10.7. We plot the eigenvalues of the individual block operators against the number of iterations needed to invert the same block operator using the GCR algorithm. In fig. 10.7a we see the values for the block Dirac operator without the weights introduced in section 9.2.4. The eigenvalues as well as the iteration counts relatively close together. Figure 10.7b, on the right, shows the eigenvalues and iteration counts for the block Dirac operator as it appears in the FETI algorithm. The average eigenvalue is about a factor three smaller, while the average iteration count is a factor 3 bigger. Furthermore, we can see that for a few blocks the eigenvalues drop significantly to values of around $\lambda_{\text{min}} = 0.01$. We indicated these points with colors in fig. 10.7b. These are the blocks that require a very large number of iterations of the GCR algorithm. The increase in the iteration count far exceeds the average factor three. The colors of these blocks match those in fig. 10.7a. We can see that before the introduction of the weights most of these blocks did not require a large number of iterations. This means that it is not, for example, the structure of the gauge fields that is responsible for the bad conditioning of the block operators, but the introduction of the weights for the FETI algorithm.

10.3.6 Deflation of the Block System

In section 8.3 we discussed the general deflation framework. Here we will work out how to use the deflation technique on the FETI block system. In the previous section we argued that exceptionally low modes of the FETI block Dirac operator correlate with large iteration numbers of the solver. In the following we consider the block with the smallest eigenvalue and correspondingly the largest number of iterations. This block is colored red in fig. 10.7 and corresponds to the red line in fig. 10.6c. The 24 lowest eigenvalues of this block are shown in fig. 10.8a. We can clearly make out one exceptionally small, real eigenvalue and a gap to the next smallest pair. This is a situation where deflation techniques are expected to work best.

In section 8.3 we argued that the projector

$$P = \sum_k \phi_k \phi_k^\dagger \tag{10.13}$$

should encompass the low modes of the operator. It is therefore useful to build the basis vectors ϕ_k from the low modes of the operator itself. Given the N smallest eigenvectors of the FETI block Dirac operator

$$D_{\text{blk}} v_k = \lambda_k v_k \tag{10.14}$$

we calculate the basis vectors ϕ_k using a Gram-Schmidt orthogonalization process[2]

$$\phi_k = \sum_{l=1}^{N} U_{kl} v_l. \tag{10.15}$$

The matrix U_{kl} is a triangular matrix defining the orthogonalization and normalization process.

The little Dirac operator E is then given by

$$E_{kl} = \phi_k^\dagger D \phi_l = \sum_{j=1}^{N} \phi_k^\dagger U_{lj} \lambda_j v_j. \tag{10.16}$$

Its inverse is efficiently calculated as

$$E_{kl}^{-1} = \phi_k^\dagger D^{-1} \phi_l = \sum_{j=1}^{N} \phi_k^\dagger U_{lj} \frac{1}{\lambda_j} v_j. \tag{10.17}$$

This operator is used in the definition of P_L and P_R and is small enough to be saved explicitly, if the number of eigenmodes is not exceedingly large.

[2] One could alternatively define the deflation basis vectors $\tilde{\phi}_k$ using the hermitian system $Q\tilde{\phi}_k = \gamma_5 D\tilde{\phi}_k = \tilde{\lambda}_k\tilde{\phi}_k$. In this case the little Dirac operator is given by $E_{kl} = \tilde{\phi}_k\gamma_5\tilde{\lambda}_l\tilde{\phi}_l$ and its inverse $E_{kl}^{-1} = \tilde{\phi}_k\frac{1}{\tilde{\lambda}_k}\gamma_5\tilde{\phi}_l$. In practice either choice results in similar convergence behavior of the block system.

Using these definitions and the ones in section 8.3, the solution is given as the sum over the components in the deflation subspace x_{dfl} which can be solved explicitly and the components in the complement x_{compl} that require the iterative solution of the deflated system.

$$x = P_R x_{\mathrm{compl}} + x_{\mathrm{dfl}}. \tag{10.18}$$

This technique requires the eigenmodes v_k of the Dirac operator to be known to substantial precision. Otherwise, the modes are not sufficiently removed from the deflated system which impedes the iterative solution of the system. One way to circumvent the necessity to calculate these modes to high precision is to use the deflation techniques as a preconditioner. One may, for example, apply the inverse Dirac operator only to the deflation subspace and use no preconditioning in the complement. This leads to the preconditioning operator

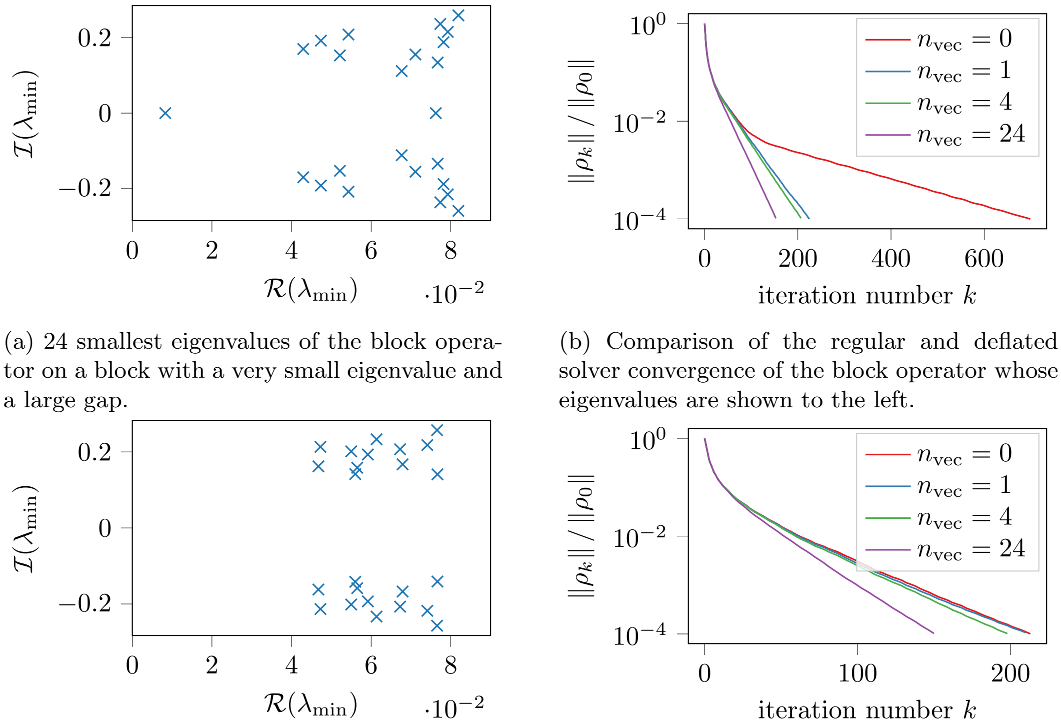
$$M_{\mathrm{dfl}} = D^{-1}P + P_L = \sum_{kl} \phi_k E^{-1}_{kl} \phi^\dagger_l + P_L. \tag{10.19}$$

Since the inverse operator E^{-1} is explicitly saved, the application of the preconditioner amounts only to a relatively small number of spinor products.

The effect of this preconditioner can be seen in fig. 10.6f. Compared to the case without any preconditioning shown in fig. 10.6c, the average number of iterations is reduced by about 35%. Additionally, the blocks with the worst convergence behavior are improved considerably and the inversion becomes much more stable. This observation is in line with the results from the previous section. A small number of very small eigenmodes spoil the convergence of the solution of the block system. Figure 10.8a shows the lower end of the spectrum of such a block with bad convergence behavior. Note the large gap between the smallest eigenvalue and the next pair. On the right, in fig. 10.8b, we show the convergence behavior of the regular GCR algorithm and the GCR algorithm with the deflation preconditioner given in eq. (10.19). The separation of the low modes seen on the left has a dramatic effect on the convergence of the iterative block solver. A large gap exists between the smallest and the other eigenvalues. For this reason deflation with a single deflation vector already has a big effect on these blocks. The inclusion of more than one deflation vector has a small effect on the convergence behavior and adds computational cost.

In figs. 10.8c and 10.8d we show the same spectrum and convergence plots for a block without a considerable gap in the eigenvalues. This situation corresponds to the majority of the blocks as seen in figs. 10.6c and 10.7b. Here the separation of the low modes does not have a large effect on the condition of the block system. Accordingly, the convergence is improved less dramatically.

Given that the dense, second spectrum shown in fig. 10.8c is more characteristic for most of the blocks, the average number of iterations per decade reduces by 44% as indicated by the last row of table 10.6. This improvement, however, is still not enough to counteract the increase in the condition number of the block system. Additionally, the additional computational cost of the preconditioner outweighs the benefits from the reduction in the number of iterations. This is in part due to suboptimal implementation.



(a) 24 smallest eigenvalues of the block operator on a block with a very small eigenvalue and a large gap.

(b) Comparison of the regular and deflated solver convergence of the block operator whose eigenvalues are shown to the left.

(c) 24 smallest eigenvalues of the block operator on a block without an exceptionally small eigenvalue and no considerable gap.

(d) Comparison of the regular and deflated solver convergence of the block operator whose eigenvalues are shown to the left.

Figure 10.8: Comparison of the spectrum (figures (a) and (c)) and convergence behavior on two different FETI blocks. Figures (b) and (d) show the relative residue for each step of the GCR solver. The lines indicate a different number of deflation vectors. The top row shows a block with an exceptionally small eigenvalue. The resulting improvement of the conversion is large if the corresponding mode is separated using the deflation technique. The bottom row shows the more common case, where the eigenvalues are larger but the spectrum is dense. The deflation is not very efficient in this case.

11 | Conclusion

The FETI algorithm is one of many in the class of domain decomposition algorithms. Domain decomposition algorithms such as the SAP have been essential in accelerating lattice QCD simulations [69, 144].

The FETI algorithm was chosen because of the prospect of improved long range convergence behavior. With the system of interface constraints, we hoped to quickly capture the coarse structure of the solution. We also anticipated improved convergence of the block system. Since the interface constraints are only enforced as the global solution converges, we expected fewer disruptions of the block system due to the contributions from neighboring blocks. Additionally, the FETI algorithm has shown competitive results in linear elasticity problems [66, 68, 157].

Despite these circumstances, in this application, we were not able to get results that compete with the current state-of-the-art solvers [70] used in lattice QCD applications. In the following, we summarize the most important differences between the FETI and the SAP algorithms and also highlight the differences in the underlying physics systems which lead to FETI's success in other applications.

FETI vs. SAP

When comparing the FETI algorithm to the SAP in lattice QCD, several key differences stand out. A first obvious remark is that the duplication of the block boundaries increases the overall problem size. This phenomenon was considered in section 10.1.1. We concluded that the increased problem size could be outweighed by the structure and conditioning of the FETI system. This leads us to additional and more severe differences.

In section 9.2.4 we discussed the FETI block operators. For the FETI system to be equivalent to the original system, we had to include weights in the block operators. In section 10.3.5 we saw that the resulting condition number of the block operators increased by an average factor of 3. We were not able to remedy this increased condition number by choosing an appropriate preconditioner for the FETI block system.

On top of that, deflation techniques that are essential to the success of the deflated SAP, did not work in the same way for the FETI algorithm. While the deflation of the FETI block system was able to enhance the convergence of the most ill-conditioned blocks, the deflation of the more important global FETI system did not have a significant effect. In section 10.2.1 we saw that the inclusion of the deflation preconditioner greatly increased the computational cost while having almost no effect on the convergence. A further study of the spectrum of the FETI operator $F_{\lambda\lambda}$ would be needed to assess if a different deflation technique is successful here.

Linear Elasticity vs. Lattice QCD

We will now compare the application of the FETI algorithm in lattice QCD to competitive applications such as the ones in [66, 68, 157]. An crucial distinction is that most competitive applications of the FETI algorithm are in two dimensions. Some extended and improved FETI type algorithms have been shown to work for three-dimensional problems [158, 161, 163]. Lattice QCD, however, is fundamentally four dimensional. With higher dimension the ratio of boundary points to the total volume scales unfavorably. Accordingly, the increase in the size of the FETI system compared to the original system is exacerbated. Additionally, the boundary is in itself a complicated, three-dimensional system. On top of that the condition number of the block system is further amplified by the smaller weights of the four dimensional system. As seen in section 9.2.4 the weights inside the block operator go down to $\frac{1}{2^d}$ with the dimension d of the system. This directly affects the condition number of the block operator.

Another important and related difference is the treatment of the block system. In two dimensions and for sufficiently small blocks it is possible to explicitly solve the block system. This can be done by using explicit methods such as the Gaussian Elimination or by storing the inverse block system acting on basis vectors. For both of these methods the condition of the block system is of secondary importance. Using explicit methods the application of the inverse block system is extremely fast and efficient. Without the computationally intensive and slow iterative solution of the block system, we could benefit from the improved convergence behavior of the FETI system seen in section 10.2.1. For four dimensional problems, however, the size of the block system is too large for explicit techniques to be realistic. In section 10.3.1 we presented the memory requirements of explicit solvers for several block sizes. Highly parallel computer architectures such as GPUs could ameliorate the computational intensity of the block system. Given the independence of individual blocks, the block system is well suited for GPUs. Further investigation that is beyond the scope of this work is needed to evaluate if the FETI algorithm is competitive on these architectures.

Acknowledgements

I acknowledge Deutsches Elektronen Synchrotron (DESY) for the infrastructure and financial support. R. Sommer was supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement nos. 813942 (ITN EuroPLEx) and 824093 (STRONG- 2020). We thank our colleagues in the Coordinated Lattice Simulations (CLS) effort [http://wiki-zeuthen.desy.de/CLS/CLS] for the joint generation of the gauge field ensembles on which the computation described here is based. We acknowledge PRACE for awarding us access to resource FERMI based in Italy at CINECA, Bologna and to resource SuperMUC based in Germany at LRZ, Munich. We acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for providing computing time through the John von Neumann Institute for Computing (NIC) on JUQUEEN at Jülich Supercomputing Centre and on SuperMUC-NG at Leibniz Supercomputing Centre (www.lrz.de). We thank DESY for computing resources on the PAX cluster in Zeuthen.

This work would not have been possible without the help and contribution from many people. In particular, I thank

Stefan Schaefer for the supervision, numerous discussions, new ideas, motivation and the review of this thesis,

Rainer Sommer for sharing his expertise, his critical questioning and his supervision at the Humboldt University,

the Zeuthen Particle Physics Theory (ZPPT) group for providing a friendly working environment,

my office colleagues, especially Manuel Schneider, Nikolai Husung and Leonardo Chimirri, for a good mix of fruitful discussions and questions as well as distractions, for chasing programming bugs and a steady supply of candy, for Choccochinos and second lunches,

my family that was always there for me and in many ways provided the foundation that made this thesis possible,

Elina for endless emotional support, for motivation and the occasional push, for always listening and being patient,

Juli, Leo, Lukas, Elina, Caddy, Domi and Manu who made me feel at home and supported me on a daily basis, for their conversations unrelated to physics and for insights into different topics,

Jakob, Linos und Ben for their support in the final and hardest phase of this thesis, for shared lunches and coffees and for refuge in their offices,

Michael Holtgrave, my high school physics teacher, who through his great classes convinced me not to drop physics in tenth grade,

my volleyball team for keeping me sane, balanced and grounded.

A Appendix

Package	Version/Commit	Author
`openQCD` [123]	1.6	Lüscher, Schaefer, Bulava, Campos, and Rago
`UWerr` [122]	Version6	Wolff
`mesons`	`8b97ab`	Korzec
`bdio` [165]	1.0/`68e915`	Korzec, Simma
`mesons-tools`	`fd8e38` with changes	Bruno
`wflow-tools`	`512e48` with minor changes	Bruno
`obs-tools-alpha`	`90beae` with minor changes	Sommer
`db-tools-alpha`	`7c45b7`	Sommer
`xml-tools-alpha`	`b11c9d`	Virotta, Sommer, Lottini, Bernardoni

Table A.1: List of software packages used in this thesis.

Bibliography

[1] Mattia Bruno, Tomasz Korzec, and Stefan Schaefer. “Setting the scale for the CLS $2+1$ flavor ensembles”. In: *Phys. Rev. D* 95.7 (2017), p. 074504. DOI: 10.1103/PhysRevD.95.074504. arXiv: 1608.08900 [hep-lat].

[2] Hang Li and P. Wang. *Solution of lepton g-2 anomalies with nonlocal QED.* Dec. 2021. arXiv: 2112.02971 [hep-ph]. Pre-published.

[3] Kenneth G. Wilson. “Confinement of Quarks”. In: *Phys. Rev. D* 10 (1974). Ed. by J. C. Taylor, pp. 2445–2459. DOI: 10.1103/PhysRevD.10.2445.

[4] Michael Creutz. “Confinement and Lattice Gauge Theory”. In: *Phys. Scripta* 23 (1981). Ed. by K. Hansen, P. Olesen, J. L. Petersen, and P. Hoyer, p. 973. DOI: 10.1088/0031-8949/23/5B/011.

[5] Michael Creutz. “QUARK CONFINEMENT”. In: *AIP Conf. Proc.* 68 (1981). Ed. by Loyal Durand and Lee G. Pondrom, pp. 296–301. DOI: 10.1063/1.32533.

[6] Michael Creutz, Laurence Jacobs, and Claudio Rebbi. “Experiments with a Gauge Invariant Ising System”. In: *Phys. Rev. Lett.* 42 (1979), p. 1390. DOI: 10.1103/PhysRevLett.42.1390.

[7] Michael Creutz. “Monte Carlo Study of Renormalization in Lattice Gauge Theory”. In: *Phys. Rev. D* 23 (1981), p. 1815. DOI: 10.1103/PhysRevD.23.1815.

[8] Kenneth G. Wilson. “MONTE CARLO CALCULATIONS FOR THE LATTICE GAUGE THEORY”. In: *NATO Sci. Ser. B* 59 (1980). Ed. by Gerard ’t Hooft et al., pp. 363–402. DOI: 10.1007/978-1-4684-7571-5_20.

[9] Y. Aoki et al. *FLAG Review 2021.* Nov. 2021. arXiv: 2111.09849 [hep-lat].

[10] S. Aoki et al. “Review of lattice results concerning low-energy particle physics”. In: *Eur. Phys. J. C* 77.2 (2017), p. 112. DOI: 10.1140/epjc/s10052-016-4509-7. arXiv: 1607.00299 [hep-lat].

[11] A. Bazavov et al. “MILC results for light pseudoscalars”. In: *PoS* CD09 (2009), p. 007. DOI: 10.22323/1.086.0007. arXiv: 0910.2966 [hep-ph].

[12] S. Durr et al. “Lattice QCD at the physical point: light quark masses”. In: *Phys. Lett. B* 701 (2011), pp. 265–268. DOI: 10.1016/j.physletb.2011.05.053. arXiv: 1011.2403 [hep-lat].

[13] S. Durr et al. “Lattice QCD at the physical point: Simulation and analysis details”. In: *JHEP* 08 (2011), p. 148. DOI: 10.1007/JHEP08(2011)148. arXiv: 1011.2711 [hep-lat].

[14] C. McNeile, C. T. H. Davies, E. Follana, K. Hornbostel, and G. P. Lepage. "High-Precision c and b Masses, and QCD Coupling from Current-Current Correlators in Lattice and Continuum QCD". In: *Phys. Rev. D* 82 (2010), p. 034512. DOI: 10.1103/PhysRevD.82.034512. arXiv: 1004.4285 [hep-lat].

[15] T. Blum et al. "Domain wall QCD with physical quark masses". In: *Phys. Rev. D* 93.7 (2016), p. 074505. DOI: 10.1103/PhysRevD.93.074505. arXiv: 1411.7017 [hep-lat].

[16] Sinya Aoki et al. "Review of Lattice Results Concerning Low-Energy Particle Physics". In: *Eur. Phys. J. C* 74 (2014), p. 2890. DOI: 10.1140/epjc/s10052-014-2890-7. arXiv: 1310.8555 [hep-lat].

[17] R. Arthur et al. "Domain Wall QCD with Near-Physical Pions". In: *Phys. Rev. D* 87 (2013), p. 094514. DOI: 10.1103/PhysRevD.87.094514. arXiv: 1208.4412 [hep-lat].

[18] A. Bazavov et al. "Up-, down-, strange-, charm-, and bottom-quark masses from four-flavor lattice QCD". In: *Phys. Rev. D* 98.5 (2018), p. 054517. DOI: 10.1103/PhysRevD.98.054517. arXiv: 1802.04248 [hep-lat].

[19] Gilberto Colangelo et al. "Review of lattice results concerning low energy particle physics". In: *Eur. Phys. J. C* 71 (2011), p. 1695. DOI: 10.1140/epjc/s10052-011-1695-1. arXiv: 1011.4408 [hep-lat].

[20] Y. Aoki et al. "Continuum Limit Physics from 2+1 Flavor Domain Wall QCD". In: *Phys. Rev. D* 83 (2011), p. 074508. DOI: 10.1103/PhysRevD.83.074508. arXiv: 1011.0892 [hep-lat].

[21] C. T. H. Davies et al. "Precise Charm to Strange Mass Ratio and Light Quark Masses from Full Lattice QCD". In: *Phys. Rev. Lett.* 104 (2010), p. 132003. DOI: 10.1103/PhysRevLett.104.132003. arXiv: 0910.3102 [hep-ph].

[22] A. Bazavov et al. "Nonperturbative QCD Simulations with 2+1 Flavors of Improved Staggered Quarks". In: *Rev. Mod. Phys.* 82 (2010), pp. 1349–1417. DOI: 10.1103/RevModPhys.82.1349. arXiv: 0903.3598 [hep-lat].

[23] C. Bernard et al. "Status of the MILC light pseudoscalar meson project". In: *PoS* LATTICE2007 (2007). Ed. by Gunnar Bali et al., p. 090. DOI: 10.22323/1.042.0090. arXiv: 0710.1118 [hep-lat].

[24] Quentin Mason, Howard D. Trottier, Ron Horgan, Christine T. H. Davies, and G. Peter Lepage. "High-precision determination of the light-quark masses from realistic lattice QCD". In: *Phys. Rev. D* 73 (2006), p. 114501. DOI: 10.1103/PhysRevD.73.114501. arXiv: hep-ph/0511160.

[25] N. Carrasco et al. "Up, down, strange and charm quark masses with $\mathrm{N}_f = 2+1+1$ twisted mass lattice QCD". In: *Nucl. Phys. B* 887 (2014), pp. 19–68. DOI: 10.1016/j.nuclphysb.2014.07.025. arXiv: 1403.4504 [hep-lat].

[26] A. T. Lytle, C. T. H. Davies, D. Hatton, G. P. Lepage, and C. Sturm. "Determination of quark masses from $\mathbf{n_f = 4}$ lattice QCD and the RI-SMOM intermediate scheme". In: *Phys. Rev. D* 98.1 (2018), p. 014513. DOI: 10.1103/PhysRevD.98.014513. arXiv: 1805.06225 [hep-lat].

[27] Bipasha Chakraborty et al. "High-precision quark masses and QCD coupling from $n_f = 4$ lattice QCD". In: *Phys. Rev. D* 91.5 (2015), p. 054508. DOI: 10.1103/PhysRevD.91.054508. arXiv: 1408.4169 [hep-lat].

[28] S. Aoki et al. “FLAG Review 2019”. In: *The European Physical Journal C* 80.2 (Feb. 2020), p. 113. ISSN: 1434-6052. DOI: 10.1140/epjc/s10052-019-7354-7. URL: https://doi.org/10.1140/epjc/s10052-019-7354-7.

[29] Mattia Bruno et al. “Light quark masses in $N_\mathrm{f} = 2 + 1$ lattice QCD with Wilson fermions”. In: *Eur. Phys. J. C* 80.2 (2020), p. 169. DOI: 10.1140/epjc/s10052-020-7698-z. arXiv: 1911.08025 [hep-lat].

[30] C. Alexandrou et al. “Quark masses using twisted-mass fermion gauge ensembles”. In: *Phys. Rev. D* 104.7 (2021), p. 074515. DOI: 10.1103/PhysRevD.104.074515. arXiv: 2104.13408 [hep-lat].

[31] Jochen Heitger, Fabian Joswig, and Simon Kuberski. “Towards the determination of the charm quark mass on $N_\mathrm{f} = 2 + 1$ CLS ensembles”. In: *PoS* LATTICE2019 (2019), p. 092. DOI: 10.22323/1.363.0092. arXiv: 1909.05328 [hep-lat].

[32] T. Blum et al. “Electromagnetic mass splittings of the low lying hadrons and quark masses from 2+1 flavor lattice QCD+QED”. In: *Phys. Rev. D* 82 (2010), p. 094508. DOI: 10.1103/PhysRevD.82.094508. arXiv: 1006.1311 [hep-lat].

[33] Marco Guagnelli, Jochen Heitger, Rainer Sommer, and Hartmut Wittig. “Hadron masses and matrix elements from the QCD Schrodinger functional”. In: *Nucl. Phys. B* 560 (1999), pp. 465–481. DOI: 10.1016/S0550-3213(99)00466-6. arXiv: hep-lat/9903040.

[34] S. Aoki et al. “2+1 Flavor Lattice QCD toward the Physical Point”. In: *Phys. Rev. D* 79 (2009), p. 034503. DOI: 10.1103/PhysRevD.79.034503. arXiv: 0807.1661 [hep-lat].

[35] A. Bazavov et al. “B- and D-meson leptonic decay constants from four-flavor lattice QCD”. In: *Phys. Rev. D* 98.7 (2018), p. 074512. DOI: 10.1103/PhysRevD.98.074512. arXiv: 1712.09262 [hep-lat].

[36] Takashi Kaneko, Brian Colquhoun, Hidenori Fukaya, and Shoji Hashimoto. “D meson semileptonic form factors in $N_f = 3$ QCD with Möbius domain-wall quarks”. In: *EPJ Web Conf.* 175 (2018). Ed. by M. Della Morte, P. Fritzsch, E. Gámiz Sánchez, and C. Pena Ruano, p. 13007. DOI: 10.1051/epjconf/201817513007. arXiv: 1711.11235 [hep-lat].

[37] W. G. Parrott, C. Bouchard, and C. T. H. Davies. *$B \to K$ and $D \to K$ form factors from fully relativistic lattice QCD.* July 2022. arXiv: 2207.12468 [hep-lat]. Pre-published.

[38] A. Bazavov et al. “$|V_{us}|$ from $K_{\ell 3}$ decay and four-flavor lattice QCD”. In: *Phys. Rev. D* 99.11 (2019), p. 114509. DOI: 10.1103/PhysRevD.99.114509. arXiv: 1809.02827 [hep-lat].

[39] A. Bazavov et al. “Kaon semileptonic vector form factor and determination of $|V_{us}|$ using staggered fermions”. In: *Phys. Rev. D* 87 (2013), p. 073012. DOI: 10.1103/PhysRevD.87.073012. arXiv: 1212.4993 [hep-lat].

[40] Nolan Miller et al. “F_K/F_π from Möbius Domain-Wall fermions solved on gradient-flowed HISQ ensembles”. In: *Phys. Rev. D* 102.3 (2020), p. 034507. DOI: 10.1103/PhysRevD.102.034507. arXiv: 2005.04795 [hep-lat].

[41] Nikolai Husung, Mateusz Koren, Philipp Krah, and Rainer Sommer. “SU(3) Yang Mills theory at small distances and fine lattices”. In: *EPJ Web Conf.* 175 (2018). Ed. by M. Della Morte, P. Fritzsch, E. Gámiz Sánchez, and C. Pena Ruano, p. 14024. DOI: 10.1051/epjconf/201817514024. arXiv: 1711.01860 [hep-lat].

[42] Mattia Bruno et al. “QCD Coupling from a Nonperturbative Determination of the Three-Flavor Λ Parameter”. In: *Phys. Rev. Lett.* 119.10 (2017), p. 102001. DOI: 10.1103/PhysRevLett.119.102001. arXiv: 1706.03821 [hep-lat].

[43] S. Aoki et al. “Precise determination of the strong coupling constant in $N_f = 2{+}1$ lattice QCD with the Schrodinger functional scheme”. In: *JHEP* 10 (2009), p. 053. DOI: `10.1088/1126-6708/2009/10/053`. arXiv: `0906.3906 [hep-lat]`.

[44] Antoine Gérardin et al. “The leading hadronic contribution to $(g-2)_\mu$ from lattice QCD with $N_\mathrm{f} = 2+1$ flavours of $\mathrm{O}(a)$ improved Wilson quarks”. In: *Phys. Rev. D* 100.1 (2019), p. 014510. DOI: `10.1103/PhysRevD.100.014510`. arXiv: `1904.03120 [hep-lat]`.

[45] D. Giusti, F. Sanfilippo, and S. Simula. “Light-quark contribution to the leading hadronic vacuum polarization term of the muon $g-2$ from twisted-mass fermions”. In: *Phys. Rev. D* 98.11 (2018), p. 114504. DOI: `10.1103/PhysRevD.98.114504`. arXiv: `1808.00887 [hep-lat]`.

[46] C. T. H. Davies et al. “Hadronic-vacuum-polarization contribution to the muon’s anomalous magnetic moment from four-flavor lattice QCD”. In: *Phys. Rev. D* 101.3 (2020), p. 034512. DOI: `10.1103/PhysRevD.101.034512`. arXiv: `1902.04223 [hep-lat]`.

[47] Sz. Borsanyi et al. “Hadronic vacuum polarization contribution to the anomalous magnetic moments of leptons from first principles”. In: *Phys. Rev. Lett.* 121.2 (2018), p. 022002. DOI: `10.1103/PhysRevLett.121.022002`. arXiv: `1711.04980 [hep-lat]`.

[48] Thomas Blum et al. “Hadronic Light-by-Light Scattering Contribution to the Muon Anomalous Magnetic Moment from Lattice QCD”. In: *Phys. Rev. Lett.* 124.13 (2020), p. 132002. DOI: `10.1103/PhysRevLett.124.132002`. arXiv: `1911.08123 [hep-lat]`.

[49] G. W. Bennett et al. “Final Report of the Muon E821 Anomalous Magnetic Moment Measurement at BNL”. In: *Phys. Rev. D* 73 (2006), p. 072003. DOI: `10.1103/PhysRevD.73.072003`. arXiv: `hep-ex/0602035`.

[50] B. Abi et al. “Measurement of the Positive Muon Anomalous Magnetic Moment to 0.46 ppm”. In: *Phys. Rev. Lett.* 126.14 (2021), p. 141801. DOI: `10.1103/PhysRevLett.126.141801`. arXiv: `2104.03281 [hep-ex]`.

[51] T. Aoyama et al. “The anomalous magnetic moment of the muon in the Standard Model”. In: *Phys. Rept.* 887 (2020), pp. 1–166. DOI: `10.1016/j.physrep.2020.07.006`. arXiv: `2006.04822 [hep-ph]`.

[52] Aurora Scapellato et al. “Proton generalized parton distributions from lattice QCD”. In: *Rev. Mex. Fis. Suppl.* 3.3 (2022), p. 0308104. DOI: `10.31349/SuplRevMexFis.3.0308104`. arXiv: `2201.06519 [hep-lat]`.

[53] Guy F. de Teramond et al. “Universality of Generalized Parton Distributions in Light-Front Holographic QCD”. In: *Phys. Rev. Lett.* 120.18 (2018), p. 182001. DOI: `10.1103/PhysRevLett.120.182001`. arXiv: `1801.09154 [hep-ph]`.

[54] Floriano; Manigrasso Floriano Manigrasso. “Direct calculation of parton distribution functions (PDFs) on the lattice”. PhD thesis. Cyprus U., Cyprus U., 2022.

[55] Rainer Sommer. “Scale setting in lattice QCD”. In: *PoS* LATTICE2013 (2014), p. 015. DOI: `10.22323/1.187.0015`. arXiv: `1401.3270 [hep-lat]`.

[56] Szabolcs Borsanyi et al. “High-precision scale setting in lattice QCD”. In: *JHEP* 09 (2012), p. 010. DOI: `10.1007/JHEP09(2012)010`. arXiv: `1203.4469 [hep-lat]`.

[57] Constantia Alexandrou et al. “Gradient flow scale-setting with $N_f = 2+1+1$ Wilson-clover twisted-mass fermions”. In: *Proceedings of The 38th International Symposium on Lattice Field Theory — PoS(LATTICE2021)* (Nov. 2021). arXiv: `2111.14710 [hep-lat]`.

[58] Roman Höllwieser, Francesco Knechtli, and Tomasz Korzec. “Scale setting for $N_f = 3+1$ QCD”. In: *Eur. Phys. J. C* 80.4 (2020), p. 349. DOI: `10.1140/epjc/s10052-020-7889-7`. arXiv: `2002.02866 [hep-lat]`.

[59] Mattia Bruno, Tomasz Korzec, and Stefan Schaefer. *Measurements of correlators and their derivatives on the ensembles H101, H102, H105, C101, N202, N203, N200, D200, N300, J303, J500.* URL: `https://wiki-zeuthen.desy.de/CLS/CLS`.

[60] Antoine Gérardin, Marco Cè, Andreas Risch, Georg von Hippel, and Daniel Mohler. *Measurements of correlators and their derivatives on the ensembles H101, H102, H105, N101, C101, B450, S400, D450, D452, N202, N203, N200, D200, E250, N300, N302, J303, E300.* URL: `https://wiki-zeuthen.desy.de/CLS/CLS`.

[61] Wolfgang Söldner, Sara Collins, and Piotr Korcyl. *Measurements of correlators on the ensembles H101, H102, H105, N101, C101, B450, S400, N202, N203, N200, D200, E250, N300, N302, J303, J500, J501.* URL: `https://wiki-zeuthen.desy.de/CLS/CLS`.

[62] Jochen Heitger, Fabian Joswig, and Simon Kuberski. *Measurements of the derivative of the action on the ensembles N300, N302, J501.* URL: `https://wiki-zeuthen.desy.de/CLS/CLS`.

[63] Tomasz Korzec and Ben Straßberger. *Measurements of correlators and their derivatives on the N202 ensemble and measurements of the derivative of the action on the ensembles N101, C101, B450, S400, D450, D452, E250, E300.* URL: `https://wiki-zeuthen.desy.de/CLS/CLS`.

[64] Martin Luscher. “Computational Strategies in Lattice QCD”. In: *Les Houches Summer School: Session 93: Modern perspectives in lattice QCD: Quantum field theory and high performance computing.* Feb. 2010, pp. 331–399. arXiv: `1002.4232 [hep-lat]`.

[65] A. Toselli and O. Widlund. *Domain Decomposition Methods - Algorithms and Theory.* Springer Series in Computational Mathematics. Springer Berlin Heidelberg, 2006. ISBN: 9783540266624. URL: `https://doi.org/10.1007/b137868`.

[66] Axel Klawonn and Olof Widlund. “A Domain Decomposition Method with Lagrange Multipliers and Inexact Solvers for Linear Elasticity”. In: *SIAM J. Sci. Comput.* 22 (Nov. 2000), pp. 1199–1219. DOI: `10.1137/S1064827599352495`.

[67] Axel Klawonn and Olof B. Widlund. “Dual-primal FETI methods for linear elasticity”. In: *Communications on Pure and Applied Mathematics* 59.11 (2006), pp. 1523–1572. DOI: `https://doi.org/10.1002/cpa.20156`.

[68] O. Widlund, Stefano Zampini, Simone Scacchi, and L. Pavarino. “Block FETI–DP/BDDC preconditioners for mixed isogeometric discretizations of three-dimensional almost incompressible elasticity”. In: *Mathematics of Computation* 90 (Feb. 2021), p. 1. DOI: `10.1090/mcom/3614`.

[69] Martin Lüscher. “Local coherence and deflation of the low quark modes in lattice QCD”. In: *JHEP* 07 (2007), p. 081. DOI: `10.1088/1126-6708/2007/07/081`. arXiv: `0706.2298 [hep-lat]`.

[70] Martin Lüscher. “Deflation acceleration of lattice QCD simulations”. In: *JHEP* 12 (2007), p. 011. DOI: `10.1088/1126-6708/2007/12/011`. arXiv: `0710.5417 [hep-lat]`.

[71] Christof Gattringer and Christian B Lang. *Quantum chromodynamics on the lattice: an introductory presentation.* Lecture Notes in Physics. Berlin: Springer, 2010. DOI: `10.1007/978-3-642-01850-3`. URL: `https://cds.cern.ch/record/1225461`.

[72] Martin Lüscher and Stefan Schaefer. "Lattice QCD without topology barriers". In: *JHEP* 07 (2011), p. 036. DOI: 10.1007/JHEP07(2011)036. arXiv: 1105.4749 [hep-lat].

[73] Martin Lüscher and Stefan Schaefer. "Lattice QCD with open boundary conditions and twisted-mass reweighting". In: *Comput. Phys. Commun.* 184 (2013), pp. 519–528. DOI: 10.1016/j.cpc.2012.10.003. arXiv: 1206.2809 [hep-lat].

[74] Rajamani Narayanan. "Running coupling in pure gauge theories using the Schrodinger functional". In: *7th Meeting of the APS Division of Particles Fields.* Dec. 1992, pp. 1484–1486. arXiv: hep-lat/9211066.

[75] Stefan Sint. "On the Schrodinger functional in QCD". In: *Nucl. Phys. B* 421 (1994), pp. 135–158. DOI: 10.1016/0550-3213(94)90228-3. arXiv: hep-lat/9312079.

[76] Achim Bode, Peter Weisz, and Ulli Wolff. "Two loop computation of the Schrodinger functional in lattice QCD". In: *Nucl. Phys. B* 576 (2000). [Erratum: Nucl.Phys.B 608, 481–481 (2001), Erratum: Nucl.Phys.B 600, 453–453 (2001)], pp. 517–539. DOI: 10.1016/S0550-3213(00)00187-5. arXiv: hep-lat/9911018.

[77] Martin Luscher. "The Schrodinger functional in lattice QCD with exact chiral symmetry". In: *JHEP* 05 (2006), p. 042. DOI: 10.1088/1126-6708/2006/05/042. arXiv: hep-lat/0603029.

[78] Mattia Bruno et al. "Simulation of QCD with $\mathrm{N}_f = 2 + 1$ flavors of non-perturbatively improved Wilson fermions". In: *JHEP* 02 (2015), p. 043. DOI: 10.1007/JHEP02(2015)043. arXiv: 1411.3982 [hep-lat].

[79] M. Pilar Hernandez. "Lattice field theory fundamentals". In: *Les Houches Summer School: Session 93: Modern perspectives in lattice QCD: Quantum field theory and high performance computing.* Aug. 2009, pp. 1–91.

[80] K. Symanzik. "Continuum Limit and Improved Action in Lattice Theories. 1. Principles and phi**4 Theory". In: *Nucl. Phys. B* 226 (1983), pp. 187–204. DOI: 10.1016/0550-3213(83)90468-6.

[81] K. Symanzik. "Continuum Limit and Improved Action in Lattice Theories. 2. O(N) Nonlinear Sigma Model in Perturbation Theory". In: *Nucl. Phys. B* 226 (1983), pp. 205–227. DOI: 10.1016/0550-3213(83)90469-8.

[82] M. Luscher and P. Weisz. "On-Shell Improved Lattice Gauge Theories". In: *Commun. Math. Phys.* 97 (1985). [Erratum: Commun.Math.Phys. 98, 433 (1985)], p. 59. DOI: 10.1007/BF01206178.

[83] B. Sheikholeslami and R. Wohlert. "Improved Continuum Limit Lattice Action for QCD with Wilson Fermions". In: *Nucl. Phys. B* 259 (1985), p. 572. DOI: 10.1016/0550-3213(85)90002-1.

[84] Martin Luscher, Stefan Sint, Rainer Sommer, and Peter Weisz. "Chiral symmetry and O(a) improvement in lattice QCD". In: *Nucl. Phys. B* 478 (1996), pp. 365–400. DOI: 10.1016/0550-3213(96)00378-1. arXiv: hep-lat/9605038.

[85] D. H. Weingarten and D. N. Petcher. "Monte Carlo Integration for Lattice Gauge Theories with Fermions". In: *Phys. Lett. B* 99 (1981), pp. 333–338. DOI: 10.1016/0370-2693(81)90112-X.

[86] P.A. Zyla et al. "Review of Particle Physics". In: *PTEP* 2020.8 (2020), p. 083C01. DOI: 10.1093/ptep/ptaa104.

[87] Peter G. Lepage. "The analysis of algorithms for lattice field theory". In: *From Actions to Answers.* WORLD SCIENTIFIC, 1990. DOI: 10.1142/0971.

[88] Nolan Miller et al. "Scale setting the Möbius domain wall fermion on gradient-flowed HISQ action using the omega baryon mass and the gradient-flow scales t_0 and w_0". In: *Phys. Rev. D* 103.5 (2021), p. 054511. DOI: 10.1103/PhysRevD.103.054511. arXiv: 2011.12166 [hep-lat].

[89] Stefano Capitani, Michele Della Morte, Georg von Hippel, Bastian Knippschild, and Hartmut Wittig. "Scale setting via the Ω baryon mass". In: *PoS* LATTICE2011 (2011). Ed. by Pavlos Vranas, p. 145. DOI: 10.22323/1.139.0145. arXiv: 1110.6365 [hep-lat].

[90] P. E. Shanahan, A. W. Thomas, and R. D. Young. "Scale setting, sigma terms and the Feynman-Hellman theorem". In: *PoS* LATTICE2012 (2012). Ed. by Derek Leinweber et al., p. 165. DOI: 10.22323/1.164.0165. arXiv: 1301.3231 [hep-lat].

[91] Robert D. Mawhinney. "Lattice QCD with zero, two and four quark flavors". In: *RHIC Summer Study 96: Brookhaven Theory Workshop on Relativistic Heavy Ions*. Nov. 1996. arXiv: hep-lat/9705030.

[92] Alan C. Irving et al. "Tuning actions and observables in lattice QCD". In: *Phys. Rev. D* 58 (1998), p. 114504. DOI: 10.1103/PhysRevD.58.114504. arXiv: hep-lat/9807015.

[93] V. G. Bornyakov et al. "Determining the scale in Lattice QCD". In: *33rd International Symposium on Lattice Field Theory*. Dec. 2015. arXiv: 1512.05745 [hep-lat].

[94] R. J. Dowdall et al. "The Upsilon spectrum and the determination of the lattice spacing from lattice QCD including charm quarks in the sea". In: *Phys. Rev. D* 85 (2012), p. 054509. DOI: 10.1103/PhysRevD.85.054509. arXiv: 1110.6887 [hep-lat].

[95] A. Gray et al. "The Upsilon spectrum and m(b) from full lattice QCD". In: *Phys. Rev. D* 72 (2005), p. 094507. DOI: 10.1103/PhysRevD.72.094507. arXiv: hep-lat/0507013.

[96] Nathan Joseph Brown. "Lattice Scales from Gradient Flow and Chiral Analysis on the MILC Collaboration's HISQ Ensembles". PhD thesis. Washington U., St. Louis, Washington U., St. Louis, 2018. DOI: 10.7936/K7S181ZQ.

[97] R. Sommer. "A New way to set the energy scale in lattice gauge theories and its applications to the static force and alpha-s in SU(2) Yang-Mills theory". In: *Nucl. Phys. B* 411 (1994), pp. 839–854. DOI: 10.1016/0550-3213(94)90473-1. arXiv: hep-lat/9310022.

[98] Marco Guagnelli, Rainer Sommer, and Hartmut Wittig. "Precision computation of a low-energy reference scale in quenched lattice QCD". In: *Nucl. Phys. B* 535 (1998), pp. 389–402. DOI: 10.1016/S0550-3213(98)00599-9. arXiv: hep-lat/9806005.

[99] Silvia Necco and Rainer Sommer. "The N(f) = 0 heavy quark potential from short to intermediate distances". In: *Nucl. Phys. B* 622 (2002), pp. 328–346. DOI: 10.1016/S0550-3213(01)00582-X. arXiv: hep-lat/0108008.

[100] Claude W. Bernard et al. "The Static quark potential in three flavor QCD". In: *Phys. Rev. D* 62 (2000), p. 034503. DOI: 10.1103/PhysRevD.62.034503. arXiv: hep-lat/0002028.

[101] Martin Lüscher. "Properties and uses of the Wilson flow in lattice QCD". In: *JHEP* 08 (2010). [Erratum: JHEP 03, 092 (2014)], p. 071. DOI: 10.1007/JHEP08(2010)071. arXiv: 1006.4518 [hep-lat].

[102] Martin Lüscher. "Future applications of the Yang-Mills gradient flow in lattice QCD". In: *PoS* LATTICE2013 (2014), p. 016. DOI: 10.22323/1.187.0016. arXiv: 1308.5598 [hep-lat].

[103] A. Bazavov et al. "Gradient flow and scale setting on MILC HISQ ensembles". In: *Phys. Rev. D* 93.9 (2016), p. 094510. DOI: 10.1103/PhysRevD.93.094510. arXiv: 1503.02769 [hep-lat].

[104] Stefan Sint and Alberto Ramos. "On O(a^2) effects in gradient flow observables". In: *PoS* LATTICE2014 (2015), p. 329. DOI: 10.22323/1.214.0329. arXiv: 1411.6706 [hep-lat].

[105] Anna Hasenfratz. "Improved gradient flow for step scaling function and scale setting". In: *PoS* LATTICE2014 (2015), p. 257. DOI: 10.22323/1.214.0257. arXiv: 1501.07848 [hep-lat].

[106] Martin Luscher. "Chiral symmetry and the Yang–Mills gradient flow". In: *JHEP* 04 (2013), p. 123. DOI: 10.1007/JHEP04(2013)123. arXiv: 1302.5246 [hep-lat].

[107] Daniel Mohler, Stefan Schaefer, and Jakob Simeth. "CLS 2+1 flavor simulations at physical light- and strange-quark masses". In: *EPJ Web Conf.* 175 (2018). Ed. by M. Della Morte, P. Fritzsch, E. Gámiz Sánchez, and C. Pena Ruano, p. 02010. DOI: 10.1051/epjconf/201817502010. arXiv: 1712.04884 [hep-lat].

[108] *Coordinated Lattice Simulations (CLS)*. URL: https://wiki-zeuthen.desy.de/CLS/CLS.

[109] John Bulava and Stefan Schaefer. "Improvement of $N_f = 3$ lattice QCD with Wilson fermions and tree-level improved gauge action". In: *Nucl. Phys. B* 874 (2013), pp. 188–197. DOI: 10.1016/j.nuclphysb.2013.05.019. arXiv: 1304.7093 [hep-lat].

[110] Daniel Mohler and Stefan Schaefer. "Remarks on strange-quark simulations with Wilson fermions". In: *Phys. Rev. D* 102.7 (2020), p. 074506. DOI: 10.1103/PhysRevD.102.074506. arXiv: 2003.13359 [hep-lat].

[111] Martin Luscher and Filippo Palombi. "Fluctuations and reweighting of the quark determinant on large lattices". In: *PoS* LATTICE2008 (2008). Ed. by Christopher Aubin et al., p. 049. DOI: 10.22323/1.066.0049. arXiv: 0810.0946 [hep-lat].

[112] Thomas A. DeGrand. "A Conditioning Technique for Matrix Inversion for Wilson Fermions". In: *Comput. Phys. Commun.* 52 (1988), pp. 161–164. DOI: 10.1016/0010-4655(88)90180-4.

[113] A. D. Kennedy, Ivan Horvath, and Stefan Sint. "A New exact method for dynamical fermion computations with nonlocal actions". In: *Nucl. Phys. B Proc. Suppl.* 73 (1999). Ed. by Thomas A. DeGrand, Carleton E. DeTar, R. Sugar, and D. Toussaint, pp. 834–836. DOI: 10.1016/S0920-5632(99)85217-7. arXiv: hep-lat/9809092.

[114] M. A. Clark and A. D. Kennedy. "Accelerating dynamical fermion computations using the rational hybrid Monte Carlo (RHMC) algorithm with multiple pseudofermion fields". In: *Phys. Rev. Lett.* 98 (2007), p. 051601. DOI: 10.1103/PhysRevLett.98.051601. arXiv: hep-lat/0608015.

[115] Martin Lüscher. "Topology, the Wilson flow and the HMC algorithm". In: *PoS* LATTICE2010 (2010). Ed. by Giancarlo Rossi, p. 015. DOI: 10.22323/1.105.0015. arXiv: 1009.5877 [hep-lat].

[116] John Bulava, Michele Della Morte, Jochen Heitger, and Christian Wittemeier. "Nonperturbative improvement of the axial current in N_f=3 lattice QCD with Wilson fermions and tree-level improved gauge action". In: *Nucl. Phys. B* 896 (2015), pp. 555–568. DOI: 10.1016/j.nuclphysb.2015.05.003. arXiv: 1502.04999 [hep-lat].

[117] Mattia Bruno. "The energy scale of the 3-flavour Lambda parameter". PhD thesis. Humboldt U., Berlin, 2015. DOI: 10.18452/17516.

[118] Mattia Dalla Brida, Tomasz Korzec, Stefan Sint, and Pol Vilaseca. "High precision renormalization of the flavour non-singlet Noether currents in lattice QCD with Wilson quarks". In: *Eur. Phys. J. C* 79.1 (2019), p. 23. DOI: 10.1140/epjc/s10052-018-6514-5. arXiv: 1808.09236 [hep-lat].

[119] Yusuke Taniguchi and Akira Ukawa. "Perturbative calculation of improvement coefficients to O(g**2a) for bilinear quark operators in lattice QCD". In: *Phys. Rev. D* 58 (1998), p. 114503. DOI: `10.1103/PhysRevD.58.114503`. arXiv: `hep-lat/9806015`.

[120] Gilberto Colangelo and Stephan Durr. "The Pion mass in finite volume". In: *Eur. Phys. J. C* 33 (2004), pp. 543–553. DOI: `10.1140/epjc/s2004-01593-y`. arXiv: `hep-lat/0311023`.

[121] Gilberto Colangelo, Stephan Durr, and Christoph Haefeli. "Finite volume effects for meson masses and decay constants". In: *Nucl. Phys. B* 721 (2005), pp. 136–174. DOI: `10.1016/j.nuclphysb.2005.05.015`. arXiv: `hep-lat/0503014`.

[122] Ulli Wolff. "Monte Carlo errors with less errors". In: *Comput. Phys. Commun.* 156 (2004). [Erratum: Comput.Phys.Commun. 176, 383 (2007)], `https://www.physik.hu-berlin.de/de/com/ALPHAsoft`, pp. 143–153. DOI: `10.1016/S0010-4655(03)00467-3`. arXiv: `hep-lat/0306017`.

[123] Martin Lüscher, Stefan Schaefer, John Bulava, Isabel Campos, and Antonio Rago. *openQCD*. Version 1.6. June 14, 2012. URL: `https://luscher.web.cern.ch/luscher/openQCD/`.

[124] Steven Weinberg. "Phenomenological Lagrangians". In: *Physica A* 96.1-2 (1979). Ed. by S. Deser, pp. 327–340. DOI: `10.1016/0378-4371(79)90223-1`.

[125] J. Gasser and H. Leutwyler. "Chiral Perturbation Theory: Expansions in the Mass of the Strange Quark". In: *Nucl. Phys. B* 250 (1985), pp. 465–516. DOI: `10.1016/0550-3213(85)90492-4`.

[126] Maarten Golterman. "Applications of chiral perturbation theory to lattice QCD". In: *Les Houches Summer School: Session 93: Modern perspectives in lattice QCD: Quantum field theory and high performance computing.* Dec. 2009, pp. 423–515. arXiv: `0912.4042 [hep-lat]`.

[127] Murray Gell-Mann, R. J. Oakes, and B. Renner. "Behavior of current divergences under SU(3) x SU(3)". In: *Phys. Rev.* 175 (1968), pp. 2195–2199. DOI: `10.1103/PhysRev.175.2195`.

[128] W. Bietenholz et al. "Tuning the strange quark mass in lattice simulations". In: *Phys. Lett. B* 690 (2010), pp. 436–441. DOI: `10.1016/j.physletb.2010.05.067`. arXiv: `1003.1114 [hep-lat]`.

[129] W. Bietenholz et al. "Flavour blindness and patterns of flavour symmetry breaking in lattice simulations of up, down and strange quarks". In: *Phys. Rev. D* 84 (2011), p. 054509. DOI: `10.1103/PhysRevD.84.054509`. arXiv: `1102.5300 [hep-lat]`.

[130] Oliver Bar and Maarten Golterman. "Chiral perturbation theory for gradient flow observables". In: *Phys. Rev. D* 89.3 (2014). [Erratum: Phys.Rev.D 89, 099905 (2014)], p. 034505. DOI: `10.1103/PhysRevD.89.034505`. arXiv: `1312.4999 [hep-lat]`.

[131] C. Allton et al. "Physical results from $2+1$ flavor domain wall QCD and SU(2) chiral perturbation theory". In: *Phys. Rev. D* 78 (11 Dec. 2008), p. 114509. DOI: `10.1103/PhysRevD.78.114509`. URL: `https://link.aps.org/doi/10.1103/PhysRevD.78.114509`.

[132] Nikolai Husung, Peter Marquard, and Rainer Sommer. "Asymptotic behavior of cutoff effects in Yang–Mills theory and in Wilson's lattice QCD". In: *Eur. Phys. J. C* 80.3 (2020), p. 200. DOI: `10.1140/epjc/s10052-020-7685-4`. arXiv: `1912.08498 [hep-lat]`.

[133] Nikolai Husung, Peter Marquard, and Rainer Sommer. "The asymptotic approach to the continuum of lattice QCD spectral observables". In: *Phys. Lett. B* 829 (2022), p. 137069. DOI: `10.1016/j.physletb.2022.137069`. arXiv: `2111.02347 [hep-lat]`.

[134] Biagio Lucini, Agostino Patella, Alberto Ramos, and Nazario Tantalo. "Charged hadrons in local finite-volume QED+QCD with C* boundary conditions". In: *JHEP* 02 (2016), p. 076. DOI: `10.1007/JHEP02(2016)076`. arXiv: `1509.01636 [hep-th]`.

[135] Sz. Borsanyi et al. "Isospin splittings in the light baryon octet from lattice QCD and QED". In: *Phys. Rev. Lett.* 111.25 (2013), p. 252001. DOI: `10.1103/PhysRevLett.111.252001`. arXiv: `1306.2287 [hep-lat]`.

[136] A. Portelli et al. "Electromagnetic corrections to light hadron masses". In: *PoS* LATTICE2010 (2010). Ed. by Giancarlo Rossi, p. 121. DOI: `10.22323/1.105.0121`. arXiv: `1011.4189 [hep-lat]`.

[137] V. G. Bornyakov et al. *Wilson flow and scale setting from lattice QCD.* Aug. 2015. arXiv: `1508.05916 [hep-lat]`. unpublished.

[138] C. Alexandrou et al. "Ratio of kaon and pion leptonic decay constants with Nf=2+1+1 Wilson-clover twisted-mass fermions". In: *Phys. Rev. D* 104.7 (2021), p. 074520. DOI: `10.1103/PhysRevD.104.074520`. arXiv: `2104.06747 [hep-lat]`.

[139] R. J. Dowdall, C. T. H. Davies, G. P. Lepage, and C. McNeile. "V_{us} from π and K decay constants in full lattice QCD with physical u, d, s and c quarks". In: *Phys. Rev. D* 88 (2013), p. 074504. DOI: `10.1103/PhysRevD.88.074504`. arXiv: `1303.1670 [hep-lat]`.

[140] Anthony Francis, Patrick Fritzsch, Martin Lüscher, and Antonio Rago. "Master-field simulations of O(a)-improved lattice QCD: Algorithms, stability and exactness". In: *Comput. Phys. Commun.* 255 (2020), p. 107355. DOI: `10.1016/j.cpc.2020.107355`. arXiv: `1911.04533 [hep-lat]`.

[141] Leonardo Chimirri and Rainer Sommer. "Investigation of the Perturbative Expansion of Moments of Heavy Quark Correlators for $N_f = 0$". In: *Proceedings of The 38th International Symposium on Lattice Field Theory — PoS(LATTICE2021)* (Mar. 2022). arXiv: `2203.07936 [hep-lat]`.

[142] Srijit Paul et al. "$I = 1$ π-π scattering at the physical point". In: *38th International Symposium on Lattice Field Theory.* Dec. 2021. arXiv: `2112.07385 [hep-lat]`.

[143] Martin Lüscher. "Schwarz-preconditioned HMC algorithm for two-flavour lattice QCD". In: *Comput. Phys. Commun.* 165 (2005), pp. 199–220. DOI: `10.1016/j.cpc.2004.10.004`. arXiv: `hep-lat/0409106`.

[144] Martin Lüscher. "Solution of the Dirac equation in lattice QCD using a domain decomposition method". In: *Comput. Phys. Commun.* 156 (2004), pp. 209–220. DOI: `10.1016/S0010-4655(03)00486-7`. arXiv: `hep-lat/0310048`.

[145] Charbel Farhat and Francois-Xavier Roux. "A method of finite element tearing and interconnecting and its parallel solution algorithm". In: *International Journal for Numerical Methods in Engineering* 32.6 (1991), pp. 1205–1227. DOI: `https://doi.org/10.1002/nme.1620320604`.

[146] Lloyd N. Trefethen and David Bau. *Numerical Linear Algebra.* SIAM, 1997. ISBN: 0-89-871361-7.

[147] Yousef Saad. *Iterative Methods for Sparse Linear Systems.* Jan. 2003. DOI: `10.1137/1.9780898718003.ch4`.

[148] G.H. Golub and C.F. Van Loan. *Matrix Computations*. 3nd. Baltimore: Johns Hopkins University Press, 1989.

[149] Magnus R. Hestenes and Eduard Stiefel. "Methods of conjugate gradients for solving linear systems". In: *Journal of research of the National Bureau of Standards* 49 (1952), pp. 409–435.

[150] Michel Verhaegen. "The minimal residual QR-factorization algorithm for reliably solving subset regression problems". In: *NASA Technical Reports* (1987).

[151] O. Axelsson. "Conjugate gradient type methods for unsymmetric and inconsistent systems of linear equations". In: *Linear Algebra and its Applications* 29 (1980). Special Volume Dedicated to Alson S. Householder, pp. 1–16. ISSN: 0024-3795. DOI: `https://doi.org/10.1016/0024-3795(80)90226-8`. URL: `https://www.sciencedirect.com/science/article/pii/0024379580902268`.

[152] O. Axelsson. "A generalized conjugate gradient, least square method". In: *Numerische Mathematik* 51.2 (Mar. 1987), pp. 209–227. ISSN: 0945-3245. DOI: `10.1007/BF01396750`. URL: `https://doi.org/10.1007/BF01396750`.

[153] Stanley C. Eisenstat, Howard C. Elman, and Martin H. Schultz. "Variational Iterative Methods for Nonsymmetric Systems of Linear Equations". In: *SIAM Journal on Numerical Analysis* 20 (1983), pp. 345–357.

[154] Peter Boyle et al. "Grid: OneCode and FourAPIs". In: *38th International Symposium on Lattice Field Theory*. Mar. 2022. arXiv: `2203.06777 [hep-lat]`.

[155] Charbel Farhat, Jan Mandel, and Francois Xavier Roux. "Optimal convergence properties of the FETI domain decomposition method". In: *Computer Methods in Applied Mechanics and Engineering* 115.3 (1994), pp. 365–385. ISSN: 0045-7825. DOI: `https://doi.org/10.1016/0045-7825(94)90068-X`.

[156] Charbel Farhat, Michel Lesoinne, Patrick LeTallec, Kendall Pierson, and Daniel Rixen. "FETI-DP: a dual–primal unified FETI method—part I: A faster alternative to the two-level FETI method". In: *International Journal for Numerical Methods in Engineering* 50.7 (2001), pp. 1523–1544. DOI: `https://doi.org/10.1002/nme.76`.

[157] Axel Klawonn and Olof Widlund. "FETI and Neumann-Neumann iterative substructuring methods: Connections and new results". In: *Communications on Pure and Applied Mathematics* 54.1 (2001), pp. 57–90. DOI: `https://doi.org/10.1002/1097-0312(200101)54:1<57::AID-CPA3>3.0.CO;2-D`.

[158] Axel Klawonn and Oliver Rheinbach. "Robust FETI-DP methods for heterogeneous three dimensional elasticity problems". In: *Computer Methods in Applied Mechanics and Engineering* 196.8 (2007). Domain Decomposition Methods: recent advances and new challenges in engineering, pp. 1400–1414. ISSN: 0045-7825. DOI: `https://doi.org/10.1016/j.cma.2006.03.023`.

[159] Roberto Molina-Sepulveda. "Hybridization of FETI Methods". PhD thesis. Paris IV: General Mathematics [math.GM]. Université Pierre et Marie Curie, Sept. 2017.

[160] Charbel Farhat and Francois-Xavier Roux. "Implicit parallel processing in structural mechanics". In: *International Association for Computational Mechanics*. 1994.

[161] Martin Kühn. "Adaptive FETI-DP and BDDC methods for highly heterogeneous elliptic finite element problems in three dimensions". PhD thesis. June 2018. DOI: `10.13140/RG.2.2.35567.12962`.

[162] Jan Mandel and Radek Tezaur. “On The Convergence Of A Dual-Primal Substructuring Method”. In: *Numerische Mathematik* 88 (May 2000). DOI: `10.1007/s211-001-8014-1`.

[163] Jan Mandel, Bedřich Sousedík, and Jakub Šístek. “Adaptive BDDC in three dimensions”. In: *Mathematics and Computers in Simulation* 82.10 (2012). "The Fourth IMACS Conference : Mathematical Modelling and Computational Methods in Applied Sciences and Engineering" Devoted to Owe Axelsson in ocassion of his 75th birthday, pp. 1812–1831. ISSN: 0378-4754. DOI: `https://doi.org/10.1016/j.matcom.2011.03.014`.

[164] Gunnar S. Bali, Vladimir Braun, Sara Collins, Andreas Schäfer, and Jakob Simeth. “Masses and decay constants of the η and η’ mesons from lattice QCD”. In: *JHEP* 08 (2021), p. 137. DOI: `10.1007/JHEP08(2021)137`. arXiv: `2106.05398 [hep-lat]`.

[165] Tomasz Korzec and Hubert Simma. *User Manual for Binary Input/Output Format bdio.* 2018. URL: `http://bdio.org/bdio.pdf`.