Third-Order Integration Schemes for the Langevin Equation and Correlated Markov Chains for Lattice QCD

Dissertation zur Erlangung des Doktorgrades der Naturwissenschaften (Dr. rer. nat.) der Fakultät für Physik der Universität Regensburg

vorgelegt von

Simon Bürger

aus

Münster

im Jahr 2022

Promotionsgesuch eingereicht am: 7. Dezember 2022
Die Arbeit wurde angeleitet von: Prof. Dr. Tilo Wettig

Abstract

The best current understanding of particle physics is formulated in the form of a quantum field theory, namely the Standard Model. In cases where perturbation theory is not applicable – such as hadron physics – calculations from first principles are provided by computer simulations of lattice QCD. The main computational task therein is the generation of random gauge fields subject to a high-dimensional probability distribution which is achieved by a variety of Monte Carlo Markov chain algorithms. One class of such algorithms is provided by numerical integration of the stochastic Langevin equation. In this work, we derive a novel third-order integration scheme for this equation and compare it to the previously best known second-order schemes. Furthermore, we investigate a novel method of exploiting correlations between multiple Markov chains run with slightly different sets of parameters. This has the potential to noticeably decrease statistical errors for a class of relevant observables.

Contents

Chapter 1

Introduction

Our current understanding of particle physics is formulated in the *standard model*, which is a quantum field theory (QFT) describing three of the four known fundamental forces of nature. To verify the validity of the standard model – and possibly gain insights into physics beyond the standard model – one must compute precise predictions from the theory as it is known. The most well-known approach to quantitative computations in quantum field theory is perturbation theory, which works well whenever the coupling constant of the interaction in question is small. At low energy scales, this is true for the electromagnetic interaction, whose strength is given by the fine-structure constant $\alpha \approx 1/137$. The strong force on the other hand is governed by the coupling constant α_s, which crucially depends on the energy scale of the process. While at large energies the coupling becomes small, known as *asymptotic freedom*, it becomes arbitrarily large at low energy scales, which leads to the *confinement* of quarks inside of hadronic bound states. So to get any quantitative prediction of the standard model for hadrons, such as nucleons, non-perturbative approaches to quantum field theory are necessary.

Lattice field theory, in particular lattice quantum chromodynamics (QCD) as introduced by Wilson [1], allows such calculations from first principles that are not limited by perturbation theory. The usual approach to lattice field theory relies on two main ideas: First, using a Wick rotation from real to imaginary time, the path integral can be interpreted as a stochastic expectation value. Second, the infinite continuous Euclidean spacetime is approximated by a discrete and finite lattice. The computational task of estimating the path integral then boils down to generating ensembles of lattice field configurations with a probability distribution proportional to e^{-S}, where S is the action of the physical theory in question, usually some lattice version of QCD.

As this probability distribution contains a huge number of correlated degrees of freedom, exact random sampling is a challenging task. It is usually achieved by constructing a Markov chain that is known to have the desired stationary distribution. Such a Markov chain can be derived in two main ways: Either using Hamiltonian dynamics or discretizing

the Langevin equation, which is a stochastic differential equation.

Hamiltonian schemes have first been considered in [2], and were later improved in [3, 4] in which form they are most commonly used today. Langevin schemes to second order were previously constructed in [5, 6, 7] and improved upon in [8]. In this thesis, we construct the first third-order integration scheme for the Langevin equation.

Any simulation results on the lattice depend on parameters such as the particular value for the coupling constant and the quark masses, which are typically chosen unphysically large. To extrapolate to physical values, it is necessary to repeat the simulation with different tuples of these parameters. So it is advantageous to not just know a particular value of a physical observable but also its dependence on the parameters of the system. In this thesis, we investigate a novel approach for such estimations that has the potential to improve statistical errors when comparing lattice results from multiple simulations in close proximity in parameter space.

This work is structured as follows: In Chapter 2 we describe the basic ideas of lattice QCD. First we derive the basic lattice action from the continuum action and then discuss some common improvements thereof. Chapter 3 introduces the statistical methods needed to analyze data coming from Monte Carlo Markov chain simulations. In particular one needs to take care of autocorrelation between successively generated gauge configurations.

Chapter 4 starts with a self-contained introduction to the mathematical theory of stochastic differential equations based on the Itô calculus and then examines the special case of the Langevin equation. A particular focus is on the different notions of convergence orders of stochastic numerical methods. The chapter culminates in the derivation of a novel third-order integration scheme applicable to the simulation of lattice gauge theories. It concludes with a quantitative comparison to previously established methods.

Finally, in Chapter 5 we investigate a novel approach to estimate the dependence of physical observables on the bare parameters of the lattice action. This exploits the correlations between two Markov chains run with different parameters but the same random noise terms. We finish with an evaluation of our method both on a toy model and on lattice QCD.

We summarize our work and give an outlook in Chapter 6.

Chapter 2

Basics of lattice QCD

We begin with a short introduction to lattice QCD, so far as it is necessary for this work. We briefly review the relevant equations from continuum QCD and then discuss all steps necessary to go from there to a properly discretized theory on a finite lattice that is suitable for numerical Monte Carlo simulations on a computer.

2.1 Continuum theory

2.1.1 Gauge theory and the Yang-Mills action

Any QFT is usually described as a set of fields representing its elementary particles and a Lagrangian, which contains all information about the propagation of fields and the interactions between them. One of the most important tools to study a QFT is its group of symmetries, i.e., the set of transformations of the fields that leave the Lagrangian invariant. Analyzing this group allows one to categorize the fields (e.g., the physical particles) in terms of representations of the symmetry group. In particular, we can assume elementary particles to transform in an irreducible representation of the group, which can be classified using the rich theory of Lie groups [9, 10].

Any field theory respecting special relativity in a 4-dimensional Minkowski space has to contain the Poincaré group as part of its symmetry. The Poincaré group is a semidirect product of the (Abelian) group of translations $\mathbb{R}^{1+3}$ and the Lorentz group $\mathrm{O}(1,3)$ containing spatial rotations and relativistic boosts. According to Wigner's classification[11], the physically relevant irreducible representations of the Poincaré group can be classified by two parameters m and s, where in physics $m \in \mathbb{R}_{\geq 0}$ is called the (rest-)*mass* of the particle, and $s \in \{0, \frac{1}{2}, 1, \frac{3}{2}, \ldots\}$ is its *spin*. Furthermore, we know from the spin–statistics theorem [12, 13] that integer spin has to correspond to bosons and half-integer spin to fermions.

Due to translational invariance and the principle of locality, the action can be written

as an integral over a local Lagrangian density, $S = \int d^4x\ L$.[1] For a single (free, Dirac) fermion field ψ for example, the Lagrangian reads

$$L[\psi] = \overline{\psi}(i\not{\partial} - m)\psi\,, \tag{2.1}$$

where $\not{\partial} = \gamma^\mu\partial_\mu$ is the "Feynman slash" notation, $\overline{\psi} = \psi^\dagger\gamma^0$ is the Dirac adjoint, and the γ^μ are the (Minkowski) Dirac matrices, which generate the Clifford algebra $\mathrm{Cl}_{1,3}(\mathbb{R})$.

This Lagrangian is manifestly relativistic invariant, i.e., it stays constant when transforming the field ψ under the Poincaré group. Gauge theory is concerned with additional local symmetries of the system. Suppose ψ transforms in some Lie group G as[2]

$$\psi(x) \to \Omega(x)\psi(x)\,, \tag{2.2a}$$
$$\overline{\psi}(x) \to \overline{\psi}(x)\Omega^\dagger(x)\,. \tag{2.2b}$$

For a Lagrangian like (2.1) to become invariant, we have to replace the derivative ∂_μ with a *covariant derivative* D_μ which transforms in the adjoint representation of G. A general ansatz[3] is $D_\mu = \partial_\mu - iA_\mu$, where A is called a *gauge field*, taking values in the Lie algebra $\mathfrak{g}$ associated with the group G. The transformation of this field under the gauge group reads

$$A_\mu(x) \to \Omega(x)A_\mu(x)\Omega(x)^\dagger - i(\partial_\mu\Omega(x))\Omega(x)^\dagger\,. \tag{2.3}$$

A quick calculation shows that this is the exact transformation law needed to make the covariant derivative D_μ transform exactly as $D_\mu \to \Omega D_\mu \Omega^\dagger$. In gauge field theory, A is now promoted to a dynamical field in its own right by adding a kinetic term for A to the Lagrangian. The usual (though not the only) choice is the *Yang-Mills* action proportional to $\operatorname{tr} F_{\mu\nu}F^{\mu\nu}$, where

$$F_{\mu\nu} = i[D_\mu, D_\nu] = \partial_\mu A_\nu - \partial_\nu A_\mu - i[A_\mu, A_\nu] \tag{2.4}$$

is the *field strength tensor*. In the mathematical terms from differential geometry, $F_{\mu\nu}$ is called the *curvature* 2-*form* of the *connection* iA_μ, and the Yang-Mills action is simply its L^2-norm. This means that critical points of the gauge action correspond to field configurations with (locally) minimal curvature.

---

[1] In a commonly accepted abuse of nomenclature we will use the terms "Lagrangian" and "Lagrangian density" interchangeably.

[2] We assume the gauge group to be compact and connected, which is true for all physically relevant cases. For such groups, all finite-dimensional representations can be assumed to be unitary, thus we use Ω^{-1} and $\Omega^\dagger$ interchangeably.

[3] The explicit factor i makes A a Hermitian matrix in the end, which is preferred in physics. In mathematics this factor is usually not present, making the gauge field anti-Hermitian.

Summarizing, the full continuum Lagrangian in the form we need is

$$L[\psi, \overline{\psi}, A] = \sum_f \overline{\psi}_f (i\not{D} - m_f)\psi_f - \frac{1}{2g^2} \operatorname{tr} F_{\mu\nu} F^{\mu\nu} \,, \tag{2.5a}$$

$$D_\mu = \partial_\mu - iA_\mu \,, \tag{2.5b}$$

$$F_{\mu\nu} = i[D_\mu, D_\nu] \,, \tag{2.5c}$$

where f labels multiple independent fermion fields called *flavors* which only differ by their (bare) mass m_f and g is the coupling constant of the gauge bosons. In continuum calculations, it is customary to rescale the gauge field as $A \to gA$, which makes it more obvious that the coupling constant g governs the interaction of the fermions. For lattice field theory though, it is more practical to leave it in this form where the coupling constant only appears as an overall factor of the gauge action.

Full QCD corresponds to the gauge group $G = \mathrm{SU}(3)$. Its gauge bosons are called *gluons* and the six fermion flavors are called *quarks*. This means that $A_\mu(x)$ are represented by traceless, Hermitian 3×3 matrices and the $\psi_f(x)$ are vector fields composed of 3 Lorentz vectors, i.e., they contain 12 (Grassmann) components per flavor. On the lattice, many other settings can be (and have been) studied. In essentially all numerical studies, only an unphysically small number of quark flavors are included in the simulation. This is justified as long as the involved energies are low enough that pair-production of the heavier quarks is suppressed.

Furthermore, other gauge groups can easily be substituted. For example, setting $G = \mathrm{U}(1)$ (with a single fermion field) one easily recovers quantum electrodynamics. It can also be useful to study $G = \mathrm{SU}(N)$ for arbitrary $N \in \mathbb{N}$ to gain some more general insights, or set $N = 2$ for a cheap-to-simulate toy model that shares many properties with real QCD. In any case, most of this work does not depend on the gauge group chosen, and in particular, the integration schemes we derive in chapter 4 are applicable to any choice of (compact) Lie group G.

2.1.2 Wick rotation and the probability distribution

The expectation value of an observable $\mathcal{O}$ in QCD (or any field theory, really) can be written as a path integral

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int D[\overline{\psi}, \psi]\, D[A]\, \mathcal{O}(\psi, \overline{\psi}, A) e^{iS[\psi, \overline{\psi}, A]} \,, \tag{2.6}$$

$$Z = \int D[\overline{\psi}, \psi]\, D[A]\, e^{iS[\psi, \overline{\psi}, A]} \,, \tag{2.7}$$

where Z is the *partition function*. In this form, the Lagrangian is not usable for numerical Monte Carlo simulations because due to the factor i the exponential can not be interpreted

as a classical probability distribution. The remedy for this is a *Wick rotation* which replaces the time dimension t with the imaginary time it. Effectively, this moves us from Minkowski space (with signature $(+,-,-,-)$) to Euclidean space (with signature $(+,+,+,+)$). The precise transformations are

$$\begin{aligned} x^0 &\to -ix_4\,, & x^j &\to x_j\,, \\ D^0 &\to iD_4\,, & D^j &\to -D_j\,. \end{aligned} \tag{2.8}$$

Carefully making these substitutions in the Lagrangian leads to the Euclidean continuum Lagrangian

$$L = \sum_f \overline{\psi}_f(\not{D} + m_f)\psi_f + \frac{1}{2g^2}\,\mathrm{tr}\, F_{\mu\nu}F_{\mu\nu}\,, \tag{2.9a}$$

$$D_\mu = \partial_\mu + iA_\mu\,, \tag{2.9b}$$

$$F_{\mu\nu} = -i[D_\mu, D_\nu] = \partial_\mu A_\nu - \partial_\nu A_\mu + i[A_\mu, A_\nu]\,, \tag{2.9c}$$

where the gamma matrices (which enter into $\not{D} = \gamma_\mu\partial_\mu$) should now satisfy the Euclidean anti-commutation relation $\{\gamma_\mu, \gamma_\nu\} = 2\delta_{\mu,\nu}$. Note that in Euclidean space it is not necessary to distinguish upper and lower indices.

The crucial point of the Wick rotation is that the exponential in the path integral (2.6) goes from e^{iS} to e^{-S}, i.e., it is now a positive real number. [4] This allows us to re-interpret the path integral as a stochastic expectation value with respect to the classical probability distribution

$$P[\psi, \overline{\psi}, A] = \frac{1}{Z}e^{-S[\psi,\overline{\psi},A]}\,, \tag{2.10}$$

where the partition function Z is an overall constant which makes the total probability equal to one.[5] The main part of this work (Chapter 4) will be concerned with methods of generating random field configurations according to this kind of distribution. The next few sections will explain various modifications to (2.10) that have to be applied before it is suitable for use in a computer simulation. The final form (which contains neither ψ nor A) can be found in Equation (2.31).

[4] Note that this relies on the fact that the Euclidean Lagrangian itself is real. In our case this is true, but for other potential terms, it is not. For example the *theta term* – which reads $\varepsilon_{\mu\nu\rho\sigma}F^{\mu\nu}F^{\rho\sigma}$ in Minkowski space – acquires a factor i in the Wick rotation, thus making e^{-S} complex again. Such terms are therefore very hard to handle in lattice simulations and we will not consider them in this work.

[5] In analytical field theory, (a generalization of) Z is crucially used as a *generating functional* containing essentially all information of the theory. In lattice field theory, however, it is just a number that never even needs to be computed explicitly.

2.2 Discretization on a lattice

The continuum Lagrangian as written in Equation (2.9) is problematic. First, it is not clear whether the path integral in the continuum is mathematically well-defined at all.[6] Second, a field with infinitely many degrees of freedom can surely not be simulated on a computer. Therefore we need to discretize the action by replacing the Euclidean space-time $\mathbb{R}^4$ with some finite lattice $\Lambda = \{0, \dots, L-1\}^4$ with the understanding that neighboring lattice sites are separated by a physical distance a called the *lattice spacing*. In this setting, any space-time integral should be replaced by a finite sum as in

$$S^{\text{cont}} = \int_{\mathbb{R}^4} d^4x \, L^{\text{cont}} \quad \rightarrow \quad S^{\text{latt}} = a^4 \sum_{n \in \Lambda} L^{\text{latt}} \,, \tag{2.11}$$

and any fields $\psi(x)$ are only defined on lattice points, i.e., $\psi(n)$ with $n \in \Lambda$. We expect to recover continuum physics in the limit $V = (aL)^4 \to \infty$ and $a \to 0$.[7] The lattice action S^{latt} is not unique, however, and many different forms have been proposed over the years. In this section, we will derive the *Wilson lattice action*, which is sufficient for this work and also serves as a basic building block for more advanced actions. Our presentation roughly follows [15, Chapter 2].

2.2.1 Gauge links and the naive fermion action

Naively, it is always possible to derive a lattice action from a continuum action by simple symbolic substitutions. For example, a derivative can be replaced by a finite difference as in

$$\partial_\mu \psi(x) \to \frac{1}{2a}(\psi(n+\mu) - \psi(n-\mu)) \,, \tag{2.12}$$

where we write $n \pm \mu$ to denote the lattice site neighboring n in the positive/negative μ direction. Indeed this approach works fine for simple cases such as a free fermion (Equation (2.1)) or a scalar field theory. For a gauge theory, however, there is a problem when trying to discretize the covariant derivative $D_\mu = \partial_\mu + iA_\mu$. In particular, there is no way to define a transformation law for the discrete field $A_\mu(n)$ under a gauge transformation $\Omega(n)$ that would lead to a proper covariant derivative that transforms in the adjoint representation of the gauge group.[8] In theory, a violation of gauge invariance in the lattice

[6] A basic theorem from functional analysis shows that there is no non-trivial Borel measure for infinite-dimensional Banach spaces. This might be circumvented by instead using Gaussian measures on abstract Wiener spaces[14] (which comes with its own set of problems), though most physicists prefer not to go into such mathematical details and be content with a purely symbolical treatment of the path integral.

[7] Generally speaking, the "thermodynamic limit" of infinite volume has to be taken before the continuum limit of vanishing lattice spacing.

[8] Practically speaking, the problem manifests in the second term of Equation (2.3). Substituting the derivative with a finite difference takes one outside the gauge algebra $\mathfrak{g}$.

action could be tolerated as long as it is recovered in the continuum limit. However, there is a far superior approach that conserves (a discrete version of) gauge invariance exactly at all lattice spacings.

In his seminal 1974 paper[1], Wilson proposed to use *gauge link variables* $U_\mu(n)$, which connect the lattice sites n and $n+\mu$ (cf. Figure 2.1), thus under a gauge rotation they should transform as

$$U_\mu(n) \to \Omega(n) U_\mu(n) \Omega(n+\mu)^\dagger \,. \tag{2.13}$$

Formally, the gauge links are related to the original field A by

$$U_\mu(n) = \mathcal{P} e^{i \int A_\nu dx_\nu} \approx e^{i a A_\mu(n)} \,, \tag{2.14}$$

where the integral is along a path connecting the lattice sites n and $n+\mu$, while $\mathcal{P}$ denotes the path-ordering operator. We emphasize that this relation is purely symbolic. In a lattice simulation, the $U_\mu(n)$, which take values in the gauge group G and not the gauge algebra $\mathfrak{g}$, are the relevant degrees of freedom, while the fields A_μ no longer appear anywhere. A major advantage of this is that G (in contrast to $\mathfrak{g}$) is compact. This means that the whole path integral has a compact domain[9] and is thus always finite. In particular, neither gauge fixing nor ghost fields are required to run the simulation or to measure true physical observables.

The covariant derivative D_μ on the lattice is defined as

$$(D_\mu \psi)(n) = \frac{1}{2a} (U_\mu(n)\psi(n+\mu) - U_{-\mu}(n)\psi(n-\mu)) \,, \tag{2.15}$$

where $U_{-\mu}(n) = U_\mu^\dagger(n-\mu)$ is the link connecting the site n to the site $n-\mu$. Using this, we can write the fermionic part of the action perfectly analogous to the continuum case as

$$\begin{aligned} S_{\text{fermion}} &= a^4 \sum_{n\in\Lambda} \overline{\psi}(n)(\not{D} + m)\psi(n) \\ &= a^4 \sum_{n\in\Lambda} \overline{\psi}(n) \left(\sum_{\mu=1}^{4} \gamma_\mu \frac{U_\mu(n)\psi(n+\mu) - U_{-\mu}(n)\psi(n-\mu)}{2a} + m\psi(n) \right) . \end{aligned} \tag{2.16}$$

This is called the *naive fermion action*, as multiple modifications have to be applied (see Sections 2.2.3 and 2.3) before it is suitable for numerical simulation.

[9] After integrating out the fermion fields as in Section 2.2.4.

2.2.2 Wilson gauge action

Finding a lattice version of the Yang-Mills gauge action is now straightforward, though somewhat tedious: Plugging the covariant derivative from Equation (2.15) into the (continuum) definition $F_{\mu\nu} = -i[D_\mu, D_\nu]$ yields a (rather long) expression for the field strength tensor in terms of products of link variables U_μ. This can then be plugged into the Yang-Mills action from Equation (2.9) to get

$$\begin{aligned} S_{\text{gauge}} &= \frac{a^4}{2g^2} \sum_{\substack{n\in\Lambda \\ \mu\neq\nu}} \operatorname{tr} F_{\mu\nu}(n)^2 \\ &= \frac{2}{g^2} \sum_{\substack{n\in\Lambda \\ \mu<\nu}} \operatorname{Re}\operatorname{tr}(\mathbb{1} - U_{\mu\nu}(n)) \qquad (2.17a) \\ U_{\mu\nu}(n) &= U_\mu(n) U_\nu(n+\mu) U_\mu^\dagger(n+\nu) U_\nu^\dagger(n)\,, \qquad (2.17b) \end{aligned}$$

where $U_{\mu\nu}$ is called the *plaquette*, which is the smallest non-trivial closed loop of link variables. Note that this combination of link variables (just any other closed loop) is invariant under the lattice gauge transformation from Equation (2.13) and taking the real part effectively averages over the two possible orientations of the plaquette due to $U_{\mu\nu}^\dagger = U_{\nu\mu}$. A pictorial representation of the situation can be found in Figure 2.1.

In a computer simulation, it is common to reparameterize the coupling constant as $\beta = \frac{6}{g^2}$, to finally arrive at the *Wilson gauge action*[1]

$$S_{\text{Wilson}} = \frac{\beta}{3} \sum_{\substack{n\in\Lambda \\ \mu<\nu}} \operatorname{Re}\operatorname{tr}(\mathbb{1} - U_{\mu\nu}(n))\,, \qquad (2.18)$$

where $\beta = \infty$ now corresponds to the free field and $\beta = 0$ to the strong coupling limit. It is noteworthy that in a simulation of pure lattice gauge theory, both of these limits can be easily accessed by setting all links to the unity matrix or to uniformly random matrices, respectively. However, neither provides much insight into the non-perturbative region of QCD, for which lattice QCD is usually employed in the first place.

2.2.3 Wilson fermions and the doubling problem

It is useful to write the fermion action in matrix form, i.e.,

$$S_{\text{fermion}} = a^4 \sum_{n,m\in\Lambda} \overline{\psi}(n) D(n,m) \psi(m)\,, \qquad (2.19)$$

where $D(n,m)$ is the *Dirac matrix*. As is common, we omit Dirac as well as color indices and only write spatial (i.e., lattice) indices as necessary. The naive fermion matrix

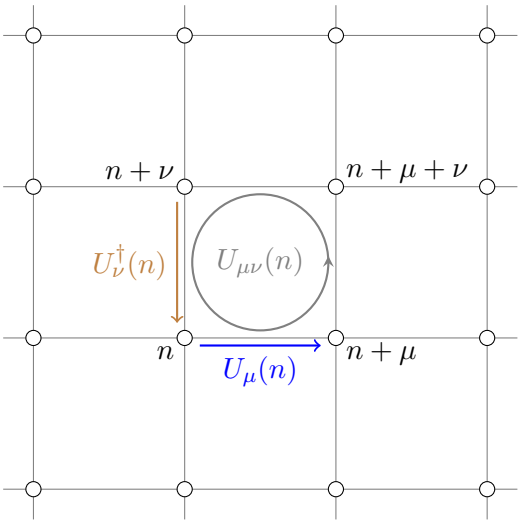

Figure 2.1: A 2-dimensional slice of the lattice in the (μ, ν) plane. Link variables connecting n to $n+\mu$ and $n+\nu$ to n are shown in blue and brown, respectively. A plaquette starting at n is shown in gray.

(Equation (2.16)) takes the form

$$D(n,m) = m\delta_{m,n} + \frac{1}{2a}\sum_{\mu=1}^{4}\gamma_\mu\left(U_\mu(n)\delta_{m,n+\mu} - U_{-\mu}(n)\delta_{m,n-\nu}\right) . \tag{2.20}$$

In the free case ($U_\mu(n) = \mathbb{1}$ for all links), one can easily compute the (discrete) Fourier transform of this operator as

$$\begin{aligned}\widetilde{D}(p,q) &= \frac{1}{|\Lambda|}\sum_{n,m\in\Lambda} e^{-ia(p\cdot n - q\cdot m)} D(n,m) \\ &= \underbrace{\frac{1}{|\Lambda|}\sum_{n\in\Lambda} e^{-ia(p-q)\cdot n}}_{=\delta(p-q)} \underbrace{\left(m + \frac{i}{a}\sum_{\mu=1}^{4}\gamma_\mu \sin(ap_\mu)\right)}_{=:\widetilde{D}(p)} .\end{aligned} \tag{2.21}$$

We see that momentum is conserved (as it must be due to translational invariance), and that in the (naive) limit $a \to 0$ the operator $\widetilde{D}(p)$ goes to $m + i\not{p}$, which is the correct continuum expression (cf. Equation (2.9)).

We can also invert $\widetilde{D}(p)$ to find the (free) quark propagator on the lattice. Using the algebra of γ matrices one can easily verify

$$\widetilde{D}(p)^{-1} = \frac{m - \frac{i}{a}\sum_\mu \gamma_\mu \sin(ap_\mu)}{m^2 + \frac{1}{a^2}\sum_\mu \sin^2(ap_\mu)} . \tag{2.22}$$

Again, the $a \to 0$ limit of this expression matches the well-known continuum propagator, $(m - i\not{p})/(m^2 + p^2)$. Looking more closely at the denominator of Equation (2.22), however, we see that the massless case has additional poles when compared to the continuum propagator. The pole at $p = 0$ corresponds to the single fermion that is in the continuum theory. But on the lattice, there are 15 additional poles whenever one or more components of p are equal to π/a and the others are equal to zero. These are referred to as *doublers* and must be removed on the lattice. Otherwise, taking the continuum limit will result in the wrong number of quark flavors.[10]

Fundamentally, the necessary existence of additional poles can be understood from the fermionic (massless) dispersion relation. Close to the origin, it must be an odd function (in order to approximate the linear continuum relation). On the lattice, however, the dispersion relation must also be periodic, so it must cross the zero axis (at least) once more. Mathematically, this is known as the *Nielsen-Ninomiya theorem*[16, 17], which roughly states that any even-dimensional, local, Hermitian, translationally-invariant, fermionic lattice theory contains the same number of left-handed and right-handed Weyl fermions. So a theory with a single chiral fermion will always produce at least one doubler. Over the years, numerous solutions to this problem have been proposed, violating different assumptions of the Nielsen-Ninomiya theorem. For example:

- Domain wall fermions[18] increase the dimension of the lattice from 4 to 5.
- Ginsparg-Wilson[19] and overlap[20] fermions obey a modified form of chiral symmetry.
- Perfect lattice fermions[21] have a non-local action.
- Twisted-mass[22] and Wilson fermions[1] explicitly break chiral symmetry.

For this work, we only consider Wilson fermions, which can be derived by adding a term to the free Dirac matrix:

$$\widetilde{D}(p) = m + \frac{i}{a}\sum_{\mu=1}^{4}\gamma_\mu \sin(ap_\mu) + \frac{1}{a}\sum_{\mu=1}^{4}(1 - \cos(ap_\mu))\,. \tag{2.23}$$

The additional *Wilson term* vanishes for the pole at $p = 0$ and increases the mass of the doublers by terms of the order $1/a$. Thus in the limit $a \to 0$, they become infinitely heavy and decouple from the theory. This modified Dirac operator is Fourier transformed back to position space and – after inserting appropriate gauge links to make the action gauge

[10] In the literature, such additional copies of fermions are often referred to as different *tastes*.

invariant – the full *Wilson fermion action* reads

$$aD(n,m) = (am+4) - \frac{1}{2}\sum_{\mu=1}^{4}((\mathbb{1}-\gamma_\mu)U_\mu(n)\delta_{m,n+\mu} + (\mathbb{1}+\gamma_\mu)U_{-\mu}(n)\delta_{m,n-\mu}) \ . \quad (2.24)$$

The explicitly broken chiral symmetry has far-reaching consequences. The renormalization of quark masses is no longer protected from a (potentially large) additive term. This means that even setting the bare quark mass to zero yields renormalized masses that are unphysically large.[11] Therefore simulations that hope to come close to the physical point have to use a negative bare quark mass. This however means that the Dirac operator is no longer protected from (approximate, or even exact) zero modes, which makes numerical inversion of the operator hard or even impossible on certain configurations of the gauge field. In the limit $a \to 0$, such *exceptional configurations* should become increasingly rare and no longer contribute to the overall average, though for practical finite values of a they pose a serious problem when trying to simulate close to physical masses. For this reason, most numerical studies with Wilson fermions are carried out with unphysically large masses, which provide some protection against exceptional configurations. Results can be extrapolated to the physical point in the end, after the simulation itself is done.

2.2.4 Integrating the fermion fields

The fermionic part of the partition function takes the form

$$Z_{\text{fermion}} = \int d\psi_N d\overline{\psi}_N \ldots d\psi_1 d\overline{\psi}_1 \ \exp\left(-\sum_{i,j}\overline{\psi}_i D[U]_{ij}\psi_j\right), \quad (2.25)$$

where we have written the Dirac operator D for a fixed gauge field U as one enormous matrix.[12] The individual components of the fermion fields are Grassmann numbers, i.e., anti-commuting. Such numbers are not directly suitable for computer simulations, but luckily, the integral in (2.25) can be solved exactly as

$$Z_{\text{fermion}} = \det(D[U]), \quad (2.26)$$

[11] Actually, on the lattice we usually only measure hadron masses, not quark masses.

[12] The fermion field generally has one color and one spin index, so there are $3 \cdot 4$ components per lattice site, though this number might be somewhat different for other kinds of lattice actions. The matrix D has therefore $144V^2$ entries, though in a very sparse pattern. In computer code, it is typically never stored explicitly.

which is known as the *Matthews-Salam formula*[23].[13] This *fermion determinant* is real but not always positive[14], so it cannot directly be used as part of a classical probability distribution. There are multiple approaches of how to deal with this, for example:

- Simply taking the absolute value of $\det(D[U])$, ignore the sign, is known as the *phase-quenched approximation*. This means the probability distribution to simulate becomes [15]

$$P_{\text{abs}}(U) = \frac{1}{Z}|\det(D[U])|e^{-S_{\text{gauge}}(U)}\,. \tag{2.27}$$

  While it is not obvious how close this approximation is to the full theory, it is a step up from the *(fully) quenched approximation* which ignores the fermion determinant altogether.

- After generating a gauge ensemble $U_i, i = 1, \dots N$ with P_{abs}, the sign of the fermion determinant can be reintroduced as a factor of any observable O. The stochastic average then reads

$$\langle O\rangle \approx \frac{\sum_{i=1}^{N}\text{sign}(\det(D[U_i]))O(U_i)}{\sum_{i=1}^{N}\text{sign}(\det(D[U_i]))}\,. \tag{2.28}$$

  This *reweighting* works fine if the proportion of configurations with a negative sign is small. Though if the proportion is close to 50%, both numerator and denominator in Equation (2.28) will be close to zero, which increases statistical noise massively, often completely drowning out any meaningful signal.

- If we have two flavors of quarks with equal mass, their combined fermion determinant can be written as

$$\det(D)^2 = \det\Big(D^\dagger D\Big) \tag{2.29}$$

  due to γ_5-hermiticity of the Dirac matrix ($\gamma_5 D\gamma_5 = D^\dagger$). This is always a non-negative number, so it can be included in the probability distribution as-is.

Most current lattice studies (including all of this work) use such degenerate quark masses in their actions for the two lightest particles, thus effectively working in the limit of unbroken isospin symmetry between up and down quarks. Additional heavier quarks can be handled with the other approaches reasonably well because sufficiently large masses

[13] It is quite easy to prove actually. The trick is that for anti-commuting numbers one has $\psi_i^2 = \overline{\psi}_i^2 = 0$, which means that most terms in the exponential vanish.

[14] This is a consequence of the explicitly broken chiral symmetry, see section 2.2.3.

[15] Note that the normalization factor Z is computed with the absolute value as well, so it is different from the "true" unquenched value.

lead to the fermion determinant being positive most of the time. The heaviest quarks are usually excluded from the simulation altogether. This is justified at low energy scales where any contributions from heavy closed quark loops are suppressed.

2.2.5 Pseudofermions

The fermion determinant D (or rather $D^\dagger D$ in the mass-degenerate case) is an important object in analytical calculations, but due to the sheer size of the matrix, it is quite impractical to keep computing during the generation of gauge configurations. The trick is to introduce an auxiliary field ϕ called a *pseudofermion* [16] with the same number of degrees of freedom as the original fermion field but taking values in the complex numbers instead of Grassmann numbers. With a simple Gaussian integral, we can write

$$\det\Big(D^\dagger D\Big) = \text{const} \cdot \int D\phi \; e^{-\phi^\dagger (D^\dagger D)^{-1} \phi}\,, \tag{2.30}$$

where the constant is absorbed into the overall normalization factor Z and each component of ϕ is integrated over the complex plane. This expression is very similar to the original action, just that the (non-Grassmann) fields are now suitable for computer simulations. The move from a fermionic to a bosonic field is effectively compensated by going from the Dirac matrix to its inverse. During the simulation, we regard ϕ as a dynamical field that is sampled in tandem with the gauge field U. Their joint probability distribution (for the case of $n_f = 2$ mass-degenerate quarks) reads

$$P[U,\phi] = \frac{1}{Z} \exp\Big(-S_{\text{gauge}}[U] - \phi^\dagger (D^\dagger[U] D[U])^{-1} \phi\Big)\,. \tag{2.31}$$

Measuring an observable on the lattice now proceeds in two steps. First, an *ensemble* of gauge configurations $U_1, \ldots, U_N$ is generated with probability distribution (2.31). Second, the expectation value of the observable is approximated by averaging its value over the ensemble,

$$\langle O \rangle \approx \frac{1}{N} \sum_{i=1}^{N} O[U_i]\,, \tag{2.32}$$

where the statistical error can usually be estimated from the variance of samples $O[U_i]$ (see Chapter 3 for more details and some caveats). Some final notes apply:

- The pseudofermion field ϕ is never stored long-term. For any fixed gauge field U it follows a simple Gaussian distribution, so a new random sample can be generated efficiently whenever necessary.

[16] Not to be confused with *noisy pseudofermions*, which are used to statistically estimate the value of $\det\big(D^\dagger D\big)$ directly.

- In Equation (2.32), the observable may only depend on the gauge field. If it also depends on the quark field(s) ψ, the fermionic part of the expectation value has to be computed analytically beforehand as in

$$O[U] = \langle O[U, \psi] \rangle_{\psi} \,. \tag{2.33}$$

  Luckily, for most observables related to hadron structure, this is easily possible using *Wick's theorem*. An example of this is shown in Equation (2.41).

- Most computational power in a typical lattice QCD simulation is used for the inversion of fermion matrices as in Equation (2.31). Therefore, the development of high-performance numerical solvers for huge sparse matrices is an important area of ongoing research.

The main part of this work (Chapter 4 and 5) is concerned with generating the ensemble itself and not with computing any particular observable.

2.3 Order a improvement

At this point, the lattice action is ready to be simulated on a computer. The remaining problem is performance. Crucially one needs to choose a range of lattice spacing a which allows reliable extrapolation to the physical $a = 0$ point. [17] This choice is constrained from both sides: If a is too large, discretization errors will be large. If a is small, the total lattice volume $V = (aL)^4$ will be small, leading to finite-volume effects.[18]

To find a balance between these two constraints, a large linear lattice size L would be necessary. As a point of reference, the CLS effort[24] is uses spatial L in the range $32 - 96$. The largest lattices used there already take about 91 GiB of storage [19], just for a single gauge field U, of which one needs thousands to do any reliable statistics. So we do not expect very much larger L to be feasible any time soon.

An elegant way of better approximating the continuum without increasing the lattice size is a systematic reduction of discretization errors. The leading-order errors for plaquette-based gauge actions are already $O(a^2)$[20] but the fermion action produces errors

[17] Technically, β is chosen as a parameter of the simulation and a is determined afterwards in a process called *scale setting*.

[18] Additionally, small lattice spacings with periodic boundary conditions can lead to a phenomenon called *topological freezing*.

[19] In CLS, the extent in the time direction is usually chosen twice as large as the spatial ones. So for $L = 96$ there are $192 \cdot 96^3 \cdot 4$ gauge links. Each gauge link is represented by a 3×3 complex matrix. Using double precision floating point numbers, the total storage thus takes $192 \cdot 96^3 \cdot 4 \cdot 3^2 \cdot 2 \cdot 8$ bytes. Other objects such as "propagators" (see Section 2.4) are even larger.

[20] Though they can be decreased further by adding larger closed loops of gauge links to the action, see for example [25].

of order $O(a)$, which should be removed whenever possible in order to improve extrapolations to $a = 0$. Following the Symanzik improvement program[26, 27], we can imagine any (local) lattice action to correspond to an effective continuum action of the form

$$S_{\text{effective}} = \int d^4x (L(x) + aL^{(1)}(x) + a^2 L^{(2)}(x) + \dots) \,, \tag{2.34}$$

where $L(x)$ is the continuum theory (Equation (2.9)) and any discretization errors are ordered by powers of a. Now any correction term in $L^{(k)}$ must be a gauge-invariant product of quark and gluon fields such that it has dimension $[L^{(k)}] = [a^{4+k}]$. For $k = 1$, the only possible continuum terms are

$$L_1^{(1)} = \overline{\psi} \sigma_{\mu\nu} F_{\mu\nu} \psi \,, \tag{2.35a}$$

$$L_2^{(1)} = \overline{\psi} \left(\overrightarrow{D}_\mu \overrightarrow{D}_\mu + \overleftarrow{D}_\mu \overleftarrow{D}_\mu \right) \psi \,, \tag{2.35b}$$

$$L_3^{(1)} = m \operatorname{tr} F_{\mu\nu} F_{\mu\nu} \,, \tag{2.35c}$$

$$L_4^{(1)} = m \left(\overline{\psi} \overrightarrow{\not{D}} \psi + \overline{\psi} \overleftarrow{\not{D}} \psi \right) \,, \tag{2.35d}$$

$$L_5^{(1)} = m^2 \overline{\psi} \psi \,, \tag{2.35e}$$

where $\sigma_{\mu\nu} = -i[\gamma_\mu, \gamma_\nu]/2$. Using the field equation $(\not{D} + m)\psi = 0$, one can derive the relations[21]

$$L_1^{(1)} - L_2^{(1)} + 2L_5^{(1)} = 0 \qquad\qquad L_4^{(1)} + 2L_5^{(1)} = 0 \,, \tag{2.36}$$

thus eliminating (for example) $L_2^{(1)}$ and $L_4^{(1)}$ from the set of operators occuring in $L^{(1)}$. Furthermore, the terms $L_3^{(1)}$ and $L_5^{(1)}$ already appear in the original action, so they can be absorbed into redefinitions of the bare parameters m and g. Thus, to cancel any $O(a)$ effects it is sufficient to add a single term proportional to $L_1^{(1)}$ to the action. We put this new term into a modified Dirac operator as

$$D_{\text{improved}} = D_{\text{Wilson}} + c_{\text{sw}} \frac{a}{2} \sum_{\mu<\nu} \sigma_{\mu\nu} \hat{F}_{\mu\nu}(n) \,, \tag{2.37}$$

where c_{sw} is called the *Sheikholeslami–Wohlert coefficient*[29] and $\hat{F}_{\mu\nu}$ is some discretized expression for the field strength tensor.

A convenient (though not unique) choice for $\hat{F}_{\mu\nu}(n)$ is as a sum of the four plaquettes in the (μ, ν) plane that touch at n:

$$\hat{F}_{\mu\nu}(n) = \frac{-i}{8a^2} (Q_{\mu\nu}(n) - Q_{\nu\mu}(n)) \,, \tag{2.38a}$$

[21] The equation of motion only proves these ralations at tree level. See [28] for proofs in the full theory.

$$Q_{\mu\nu}(n) = U_{\mu\nu}(n) + U_{\nu,-\mu}(n) + U_{-\mu,-\nu}(n) + U_{-\nu,\mu}(n)\,. \tag{2.38b}$$

Due to the shape of these four plaquettes, the last term in Equation (2.37) is often referred to as *clover improvement*. For two flavors of Wilson fermions, an appropriate value for the coefficient in the range $g \in [0, 1.1]$ is

$$c_{\text{sw}} = \frac{1 - 0.454g^2 - 0.175g^4 + 0.012g^6 + 0.045g^8}{1 - 0.720g^2}\,, \tag{2.39}$$

which was obtained non-perturbatively in [30].[22] This is the value used in the numerical results of chapter 4. Note that a different value for c_{sw} has to be used when any part of the action (such as the gluon action or the number of quarks) is changed.

2.4 Hadron spectroscopy

In this section, we will briefly summarize how to extract physical observables from lattice data. In this, we will stick to hadron masses, which are both easy to measure on the lattice[23] and have a clear physical meaning without any explicit renormalization. The hadronic 2-point correlation functions from which hadron masses are estimated also enter into the determination of more advanced observables like hadronic matrix elements.

2.4.1 Interpolators and symmetries

The first step in hadron spectroscopy is choosing appropriate *interpolators* $O, \overline{O}$ such that their corresponding Hilbert space operators $\widehat{O}, \widehat{O}^\dagger$ annihilate and create the particle state(s) we want to analyze. On the lattice, this means essentially any gauge-invariant combination of quark and gluon fields. The basic building blocks for meson interpolators are completely local operators multiplying two quark fields. The simplest pseudoscalar meson operator reads

$$O(n) = \overline{\psi}_d(n)\gamma_5\psi_u(n)\,, \tag{2.40}$$

where n is a single lattice site and u and d label the two lightest quark flavors.

A quick calculation shows that this operator transforms with a positive sign under charge conjugation and with a negative sign under parity (reflection of the three spatial dimensions). The lightest physical meson with these properties (and matching quark

[22] A perturbative computation of c_{sw} is also possible, for example Sheikholeslami[29] originally computed $c_{\text{sw}} = 1 + 0.2659g^2 + O(g^4)$. This, however, leaves discretization errors of order $O(ag^4)$ untouched, so the non-perturbative approach is generally preferred nowadays.

[23] Of course, as with all numerical and statistical methods, one can employ an arbitrarily advanced analysis to achieve maximum accuracy, see for example [31]. For this work, a very basic analysis is sufficient.

content) is the pion[24], so Equation (2.40) is also called the pion interpolator. Though it should be noted that due to all the approximations done (unphysical masses, discretization artifacts, broken Lorentz symmetry, and many more), the relation between a physical pion and this lattice operator can be somewhat loose. In particular, due to the absence of electroweak interactions, the lattice pion is completely stable and does not decay.

By choosing different combinations of quark flavors and gamma insertions in Equation (2.40), all other mesons can be created. We stick to the pion because it is the lightest one.

2.4.2 Correlation functions on the lattice

The subject of interest for us are 2-point correlation functions $\langle O(n)\overline{O}(m)\rangle$. Here the angle brackets denote an expectation value with respect to all dynamical degrees of freedom, i.e., both gluons and fermions. As the latter ones have been integrated out for the lattice simulation (see section 2.2.4), we need to compute the fermionic expectation value analytically. For the pion interpolator (2.40) this is easily done using *Wick's theorem*,

$$\begin{aligned}\langle O(n)\overline{O}(m)\rangle_\psi &= \langle\overline{\psi}_d(n)\gamma_5\psi_u(n)\overline{\psi}_u(m)\gamma_5\psi_d(m)\rangle_\psi \\ &= \mathrm{tr}\left(\gamma_5 D_u^{-1}(n,m)\gamma_5 D_d^{-1}(m,n)\right) \\ &= \mathrm{tr}\left(D_u^{-1}(n,m)D_d^{-1\dagger}(n,m)\right)\,, \end{aligned} \tag{2.41}$$

where $D_{u,d}$ are the Dirac operators of the quarks[25], the trace is with respect to color and spin indices, and we used γ_5-hermiticity ($\gamma_5 D\gamma_5 = D^\dagger$) in the last line. On the lattice, this *point-to-point correlator* is computed by numerical inversion of the Dirac operator on a given gauge field configuration and the gluonic expectation value is implemented by stochastic means as averaging over many gauge configurations.

While in principle, the point-to-point correlator does contain all information, in practice it is advantageous to do a Fourier transform of the spatial components in order to fix the total momentum of the particle. When computing a 2-point correlator, it is sufficient to fix the momentum at one end, typically the annihilation operator (also called *sink*). The creation operator (also called *source*) can be kept at a single lattice point, typically

[24] Strictly speaking, Equation (2.40) corresponds to the π^+ particle. Though as lattice works in exact isospin symmetry (see section 2.2.5), there is rarely a need to distinguish between the three physical pion particles.

[25] In our mass-degenerate case, these two are equal, the indices here are just kept for clarity.

the origin[26]. Writing 3-vectors with an arrow, we define the *2-point function*

$$C_{\text{2pt}}^{(\vec{p})}(n_t) = \sum_{\vec{n}} e^{-i\vec{n}\cdot\vec{p}} \langle O(\vec{n}, n_t)\overline{O}(\vec{0}, 0)\rangle \,, \tag{2.42}$$

where $\vec{p} \in \{0, \ldots, L-1\}^3$ is any fixed lattice momentum. In order to evaluate Equation (2.42) on a single gauge configuration, one has to solve 12 linear equations of the form $D\Psi = S^{(\alpha,a)}$, one for each spin component α and color component a. Here, the solutions Ψ are called *propagators*. Note that the source-momentum $\vec{p}$ (and also the gamma insertion in the interpolator) can be varied without needing to redo the numerical solving. For our form of the correlator, S is simply the *point source*

$$S(m)_{\beta,b}^{(\alpha,a)} = \delta_{m,0}\delta_{\alpha\beta}\delta_{ab} \,, \tag{2.43}$$

with exactly one non-zero entry.

2.4.3 Spectral decomposition

On the physical side, a correlation function is understood as a vacuum expectation value. Physically allowed states can then be observed in its spectral decomposition

$$\begin{aligned} C_{\text{2pt}}^{(\vec{p})}(n_t) &= \langle O(n_t)\overline{O}(0)\rangle \\ &= \sum_k \langle 0|\hat{O}|k\rangle\langle k|\hat{O}^\dagger|0\rangle e^{-an_tE_k} \\ &= A_0 e^{-an_tE_0} + A_1 e^{-an_tE_1} + \ldots \,, \end{aligned} \tag{2.44}$$

where the sum k is over all states in the system (with matching quantum numbers), and $E_0 < E_1 < \ldots$ are their energies. Thanks to the Euclidean spacetime, the exponent is real, so that for large n_t, one can extract the ground state energy aE_0, which is related to the rest-mass of the particle in question. The fitting strategy for this sum of exponentials will be discussed in Section 3.2.3.

It should be noted that there is a great deal of freedom in choosing the source vector S from Equation (2.43). As long as it is localized to a few (usually only one) time slices, the energy levels E_k will not be affected. In the literature, many alternatives (so-called *smeared* sources) have been suggested, for an overview see [32]. Smeared sources can be used to suppress contributions from excited states by increasing the overlap A_0 relative to the other A_i at the expense of an increase in overall statistical noise. In chapters 4 and 5, we are more interested in statistical precision than in minimizing systematic uncertainties,

[26] This is only true in theoretical calculations. In practical code, the source is moved to multiple randomized positions on the lattice, keeping the source-sink separation constant. Due to translational symmetry, this is allowed and can potentially decrease statistical errors. Results are always presented with the source shifted to the origin.

therefore we will stick to point sources as presented here.

For simple cases, such as the pion, the ground state energy is given by $E_0 = \sqrt{m^2 + |\vec{p}|^2}$, where $\vec{p}$ is the three-momentum at the sink, and all further E_i represent excited states of the same particle. For this work, we are only interested in the rest mass m. Figure 2.2 shows a typical example of a two-point correlator computed on the lattice.

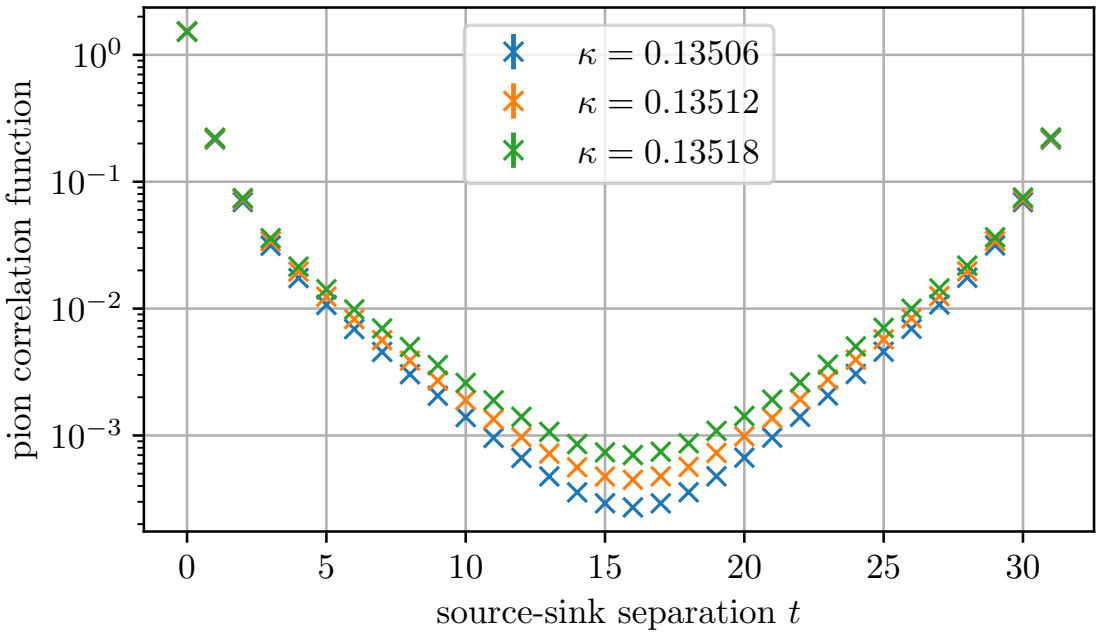

Figure 2.2: Pion correlator for momentum $\vec{p} = 0$ measured on a $16^3 \cdot 32$ lattice. The different data sets correspond to different values of the bare quark mass $am_q = \frac{1}{2\kappa} - 4$. The symmetry is due to the periodic boundary conditions. The error bars are smaller than the symbols.

Chapter 3

Basics of statistical methods

In this chapter, we will discuss some basics of statistical methods for analyzing data coming from Monte Carlo simulations. While nothing in this chapter is new, we opt to present it here in some detail. Especially the finer points in the analysis of autocorrelation (Section 3.1.3) are sometimes glossed over in the literature on lattice QCD. Furthermore, the problem of fitting a sum of exponentials to noisy data is notoriously unstable and thus warrants some discussion in Section 3.2.3. All ideas discussed here are used for analyzing the merit of the novel methods proposed in this work in Section 4.6 and Chapter 5. As the goal of this work is not to present a new or improved physical result but instead to focus on methodology, precise statistical evaluation is of particular importance.

3.1 Statistical estimators

Let X be a random variable. The goal of statistical estimators is to get an approximate value of some statistic $\theta(X)$ using only a finite sample $x_1, ..., x_n$ drawn from the distribution of X. For one-dimensional distributions, θ can simply be the mean or some higher moment. For high-dimensional X, such as the gauge configurations in lattice QCD, we are usually interested in more complicated functions of X such as hadron masses or other physical observables. In any case, we assume that θ itself is a scalar value (though we might be interested in estimating multiple different quantities using a single sample). Now let $\hat{\theta}$ be an arbitrary estimator.

1. The estimator is *consistent* if, in the limit of large sample sizes, the estimated value converges to the true value with probability one (*almost surely*), i.e.,

$$\hat{\theta}\big((x_i)_{i=1}^n\big) \xrightarrow[n\to\infty]{\text{a.s.}} \theta(X)\,. \tag{3.1}$$

2. The estimator is *unbiased* if the expectation value of the estimator is exactly equal

to the true value, even for finite sample sizes, i.e.,

$$\left\langle \hat{\theta}\big((x_i)_{i=1}^n\big) \right\rangle = \theta(X)\,. \tag{3.2}$$

If this is not the case, it is conventional to analyze the bias in orders of $\frac{1}{n}$ as in

$$\left\langle \hat{\theta}\big((x_i)_{i=1}^n\big) \right\rangle = \theta(X) + \frac{1}{n}B(X) + O\left(\frac{1}{n^2}\right)\,, \tag{3.3}$$

where B is some secondary statistic of X. In Section 3.1.4 we will discuss a generic method of estimating and even removing this leading order bias term for any given consistent estimator.

3. The *error* of the estimator is defined as

$$\mathrm{Err}^2_{\hat{\theta}} := \left\langle (\hat{\theta} - \theta)^2 \right\rangle\,, \tag{3.4}$$

i.e., the expected (squared) deviation from the true value. Note that only in the case of an unbiased estimator this is equal to the variance of $\hat{\theta}$.

In the following subsections, we will discuss a few such estimators that we need in order to evaluate the numerical simulations in Chapter 4 and 5.

3.1.1 Uncorrelated sample

As a warmup, consider a sample of n uncorrelated data points x_i from a distribution with mean μ and covariance σ^2, i.e.,

$$\langle x_i \rangle = \mu\,, \tag{3.5a}$$

$$\mathrm{Cov}(x_i, x_j) = \delta_{i,j}\sigma^2\,. \tag{3.5b}$$

It is easy to see that the sample mean,

$$\overline{x} := \frac{1}{n}\sum_{i=1}^n x_i\,, \tag{3.6}$$

is an unbiased estimator of the true mean μ. The error of this estimator is easy to calculate,

$$\mathrm{Err}^2_{\overline{x}} := \mathrm{Var}(\overline{x}) = \frac{1}{n^2}\sum_{i,j=1}^n \mathrm{Cov}(x_i, x_j) = \frac{\sigma^2}{n}\,. \tag{3.7}$$

Estimating the variance σ^2 (in order to get an error estimate for the mean) is slightly

more involved. The expectation value of the sample variance is

$$s^2_{\text{naive}} := \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2 \tag{3.8a}$$

$$\begin{aligned}\left\langle s^2_{\text{naive}}\right\rangle &= \frac{1}{n}\sum_{i=1}^{n}\left\langle (x_i-\overline{x})^2\right\rangle = \frac{1}{n}\sum_{i=1}^{n}\text{Var}(x_i-\overline{x}) \\ &= \frac{1}{n}\sum_{i=1}^{n}(\sigma^2 + \text{Var}(\overline{x}) - 2\ \text{Cov}(x_i,\overline{x})) \\ &= \frac{n-1}{n}\sigma^2 \end{aligned} \tag{3.8b}$$

This means that the naive estimator is consistent but has a small bias $\frac{n-1}{n}$, which is called the "Bessel factor". An unbiased estimator of the variance [27] can thus simply be obtained as

$$s^2 := \frac{n}{n-1}s^2_{\text{naive}} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\overline{x})^2\,. \tag{3.9}$$

3.1.2 Sample with autocorrelation

If the sample points x_i are not drawn independently, the error estimates become more involved. Assuming the data was generated by a stationary Markov process – such as the schemes discussed in Chapter 4 – the covariance between two data points then only depends on their distance, i.e.,

$$\langle x_i\rangle = \mu\,, \tag{3.10a}$$

$$\text{Cov}(x_i,x_j) = C_{|i-j|}\,, \tag{3.10b}$$

where $C_0 = \sigma^2$ is the (true) variance of the process just like before, and the *autocorrelation coefficients* C_t tend to zero for large t. The sample mean $\overline{x}$ is still an unbiased estimator of the mean μ. But the error of this estimator now is

$$\text{Err}^2_{\overline{x}} = \text{Var}(\overline{x}) = \frac{1}{n^2}\sum_{i,j=1}^{n}C_{|i-j|} = \frac{1}{n}\sigma^2 + \frac{2}{n^2}\sum_{t=1}^{n-1}(n-t)C_t \approx 2\,\tau_{\text{int}}\frac{\sigma^2}{n}\,, \tag{3.11}$$

[27] Note that this is only an unbiased estimator of the variance. The square root of this is not an unbiased estimator of the standard deviation. Computing a precise correction factor for the latter is surprisingly difficult, see for example [33].

where we defined the *integrated autocorrelation time* [28] as

$$\tau_{\text{int}} := \frac{1}{2} + \sum_{t=1}^{\infty} \frac{C_t}{C_0} \,. \tag{3.12}$$

This autocorrelation time is now a property of the Markov process that was used to generate the samples x_i. So in order to get an error estimate we now need an estimator for both σ^2 and τ_{int}. Using a similar approximation to the previous computation, we can check that

$$s^2 := \frac{1}{n - 2\,\tau_{\text{int}}} \sum_{i=1}^{n} (x_i - \overline{x})^2 \tag{3.13}$$

is a (nearly) unbiased estimator of the variance. Both Equations (3.11) and (3.13) suggest that an (auto-)correlated sample of size n is statistically equivalent to an uncorrelated sample of size $n/(2\,\tau_{\text{int}})$. This relation will be crucial in Chapter 4 where we compare numerical algorithms that sample from the same distribution but with potentially different autocorrelation times. The estimation of τ_{int} is a much more involved topic and is discussed in the next section.

3.1.3 Estimating the autocorrelation time

The integrated autocorrelation time of a time series x_i is given by

$$\tau_{\text{int}} := \frac{1}{2} + \sum_{t=1}^{\infty} \frac{C_t}{C_0} \,, \tag{3.14a}$$

$$C_t := \langle (x_i - \mu)(x_{i+t} - \mu) \rangle \,. \tag{3.14b}$$

The simplest method of estimating τ_{int} is to directly replace all quantities in the definition by their sample counterparts, i.e.,

$$\hat{C}_t := \frac{1}{n - t} \sum_{i=1}^{n-t} (x_i - \overline{x})(x_{i+t} - \overline{x}) \,. \tag{3.15}$$

Similar to Equation (3.13), the bias of $\hat{C}_t$ can be reduced greatly by replacing the prefactor $(n-t)^{-1}$ by $(n - 2\tau_{\text{int}}\, t)^{-1}$, though that might not be feasible in practice as it makes the estimation of τ_{int} recursive. The bigger problem comes from the fact that the estimator $\hat{C}_t$ becomes very noisy for large t. Therefore, in lattice QCD autocorrelation is typically

[28] Some authors absorb the factor of 2 into the definition of τ_{int}.

estimated by a truncated formula such as

$$\hat{\tau}_{\text{int}} = \frac{1}{2} + \sum_{t=1}^{t_{\max}} \frac{\hat{C}_t}{\hat{C}_0}\,, \tag{3.16}$$

with $t_{\max} \approx 5 \cdot \tau_{\text{int}}$ determined recursively. This truncation introduces a bias that can not be easily removed. In particular, this $\hat{\tau}_{\text{int}}$ is not even consistent in the $n \to \infty$ limit and will generally underestimate the true autocorrelation, which due to Equation (3.11) leads to a systematic underestimation of essentially all statistical errors coming from data generated by a Markov chain.

3.1.4 Bootstrap method

Let X be a random variable, $\theta(X)$ some statistical quantity, and $\hat{\theta}$ a consistent estimator of θ based on a finite sample $x = (x_1, \ldots, x_n)$ drawn from the distribution of X. Generally (as we already saw in the case of $\theta =$ variance), this estimator can have a bias, i.e.,

$$\langle \hat{\theta}(x) \rangle_x = \theta(X) + \frac{1}{n} B(X) + O\left(\frac{1}{n^2}\right) \tag{3.17}$$

where $B(x)$ is some statistical quantity describing the leading-order bias.

The Bootstrap-Method is a *resampling* method, in which we generate a large number of artificial samples by randomly drawing (with replacement) n elements $y_1, \ldots, y_n$ of the base sample $x_1, \ldots, x_n$. Then we evaluate $\hat{\theta}(y)$ on each of these samples and analyze the distribution of estimated values:

$$\hat{\theta}_{\text{bs}}(x) := \langle \hat{\theta}(y) \rangle_y = \theta(x) + \frac{1}{n} B(x) + O(n^{-2})\,, \tag{3.18a}$$

$$\begin{aligned} \langle \hat{\theta}_{\text{bs}}(x) \rangle_x &= \langle \hat{\theta}(x) \rangle_x + \frac{1}{n} \langle B(x) \rangle_x + O(n^{-2}) \\ &= \theta(X) + \frac{2}{n} B(X) + O(n^{-2})\,. \end{aligned} \tag{3.18b}$$

So this "mean of bootstrap-samples" is again an estimator for $\theta(X)$ which can be used for two purposes:

1. By combining these two different estimators, we can construct an improved estimator

$$\hat{\theta}_{\text{improved}}(x) := 2\hat{\theta}(x) - \hat{\theta}_{\text{bs}}(x)\,, \tag{3.19a}$$

$$\langle \hat{\theta}_{\text{improved}}(x) \rangle_x = \theta(X) + O(n^{-2})\,, \tag{3.19b}$$

which does not have a bias at the $O(\frac{1}{n})$ level. In principle, this procedure can be

repeated, creating "bootstrap-of-bootstrap" samples, which allows removing higher-order biases. But in practice that becomes computationally very expensive (and possibly noisy) and is thus seldom used. We will stick to the one-level bootstrap.

2. The variance of bootstrap samples can be used to get an error estimate of $\hat{\theta}(x)$ because

$$\mathrm{Var}_y(\hat{\theta}(y)) = \mathrm{Var}_x(\hat{\theta}(x)) + O(n^{-1}) = \mathrm{Err}^2_{\hat{\theta}} + O(n^{-1})\,. \tag{3.20}$$

In practice, the improved estimator is rarely used directly. Instead, the difference between $\hat{\theta}$ and $\hat{\theta}_{\mathrm{bs}}$ is interpreted as a systematic uncertainty. If it is small (compared to statistical errors), it is irrelevant. And if it is large, one generally tries to switch to a different estimator. The error estimate of Equation (3.20) however is ubiquitous in lattice QCD because it works with arbitrary complicated $\hat{\theta}$. In particular, it enables one to compute the error of an estimator that is itself a complicated fit, such as the ones discussed in the next section.

3.2 Model fitting

In this section, we will discuss the problem of extracting a model (or the parameters of a fixed class of models) from noisy data, also called *fitting*. After a brief discussion of *maximum likelihood estimation* (MLE) on which most data analysis in lattice QCD is based, we will present the general formulas for fitting a linear model (Section 3.2.2) as well as an approach to a certain form of non-linear fitting (Section 3.2.3). Both of these will be used in Section 5 to analyze our results.

3.2.1 Maximum likelihood estimation

Consider a physical system (or in our case a simulation thereof) governed by a set of parameters θ and resulting in some observed data x. Our goal is to find an estimate θ based on x, as well as some estimate of uncertainty. Ideally, we would like to find the parameter value which is most likely given the data. Mathematically we might write this as

$$\hat{\theta} = \arg\max_{\theta}\ \Pr(\theta \,|\, x)\,. \tag{3.21}$$

Using *Bayes' theorem* we can rewrite this as

$$\hat{\theta} = \arg\max_{\theta}\ \frac{\Pr(x \,|\, \theta)\Pr(\theta)}{\Pr(x)}\,, \tag{3.22}$$

where $\Pr(x \mid \theta)$ is the *likelihood* of observing the data x given the parameters θ. If we assume a particular model for our physical system, this is a known function. The denominator $\Pr(x)$ can generally be ignored as is independent of θ, and $\Pr(\theta)$ is called the *prior probability* of the parameter value θ. In Bayesian statistics, it represents one's belief about the probable values of θ before taking into account the current observations. In this way, Bayes' theorem should be understood as a rule for updating one's belief about the probability distribution of θ after an observation.

Fundamentally, it is impossible to fix a prior distribution to start with in a completely objective way. For this reason, many researchers in physics prefer to "use no priors", i.e., neglect the term in Equation (3.22), which leads to

$$\hat{\theta} = \arg\max_{\theta} \Pr(x \mid \theta)\,, \tag{3.23}$$

which is called the *maximum likelihood estimator* (MLE) for the parameter θ. It can be shown[29] that the MLE is generally a consistent estimator (i.e., for large sample sizes it converges to the true value) but is generally not unbiased. For example, the MLE for the variance of a Gaussian variable is equal to the variance of the sample, which is (as seen in Section 3.1.1) not an unbiased estimator for the true variance. Another famous example is a simple uniform distribution $U(0, \theta)$, for which the MLE will systematically underestimate the upper bound θ. [30]

In the literature, the maximum likelihood estimator is typically written as

$$\hat{\theta} = \arg\max_{\theta} \ell(\theta)\,, \tag{3.24}$$

where $\ell(\theta) = \log L(\theta) = \log \Pr(x \mid \theta)$ is the *log-likelihood* function. In the next section, it will become clear why taking the logarithm here is advantageous for practical calculations.

3.2.2 General linear fit

Suppose we have measured some data (t_i, y_i), $i = 1, \ldots, n$ which we assume should follow a function of the form

$$\varphi(a; t) = a_1\varphi_1(t) + a_2\varphi_2(t) + \ldots + a_m\varphi_m(t) \tag{3.25}$$

with known functions φ_j. The problem of estimating the unknown coefficients a_j is known as a *linear fit*. Assuming the error of the data y_i to be independent and identically normally

[29] Making some assumptions about the problem such as continuity and log-convexity of the likelihood function. For more details, we refer to standard textbooks on statistical interference like [34] and [35].

[30] Colloquially, this is called the "German tank problem"[36] after its historical application in World War II.

distributed[31], the log-likelihood function will be (up to irrelevant constants)

$$\ell(a) = \sum_{i=1}^{n} \left(\varphi(a; t_i) - y_i\right)^2 . \tag{3.26}$$

I.e., we get the method of least squares where the squares stem from the form of the (assumed) Gaussian distribution. This method can be improved if we furthermore know the size of the (statistical) errors of all the y_i and the correlations between them. Suppose all covariances $C_{ij} = \mathrm{Cov}(y_i, y_j)$ are known, then the likelihood function becomes

$$\begin{aligned} \ell(a) &= \sum_{i=1}^{n} \sum_{j=1}^{n} (\varphi(a; t_i) - y_i) C_{ij} (\varphi(a; t_j) - y_j) \\ &= (Ma - y)^T C (Ma - y) , \end{aligned} \tag{3.27}$$

where we defined the matrix $M_{ij} = \varphi_j(t_i)$. The covariance matrix C is always symmetric positive definite, so this is a convex optimization problem. The unique [32] solution to this problem can be found easily with basic linear algebra. We can also write it concisely as

$$a = BC^{-1/2} y , \quad \text{where } B = (C^{-1/2} M)^+ , \tag{3.28}$$

and $(\cdot)^+$ denotes the *Penrose pseudoinverse* matrix. Two final notes apply:

1. In practice, the covariance matrix C is usually not known a priori and has to be estimated from noisy data. Any inaccuracy of C is vastly amplified when computing the inverse. Therefore, one often only considers the diagonal part of C by fixing all or most off-diagonal elements to zero. In general, this introduces a bias to the fit but is often required for overall stability.

2. In principle, error bars (and correlations) of the fitted parameters a can be computed using the formula

$$\mathrm{Cov}(a) = BB^T . \tag{3.29}$$

   But this should only be used in very simple cases. For all use cases in this work, we compute statistical errors of fitting parameters using the much more robust bootstrap method (see Section 3.1.4), which also gives us estimates of any biases that might have been introduced by truncating C or by potential non-Gaussian errors in the original data.

[31] Even though in the real world, data is rarely exactly normally distributed, it is often a reasonable approximation if no other theory is available. This is due to the central limit theorem in case each data point y_i is itself accumulating errors from many separate measurements.

[32] We can always assume that M has full rank, otherwise the different components φ_j of the model are not independent.

3.2.3 Variable projection method for fitting sums of exponentials

One particular type of fit that is often necessary for lattice QCD is a sum of exponential functions of the form

$$\varphi(a_0, \ldots, a_n, \lambda_1, \ldots, \lambda_n; t) = a_0 + a_1 e^{-\lambda_1 t} + \ldots + a_n e^{-\lambda_n t}\,, \tag{3.30}$$

in which both the prefactors a_i as well as the decay constants λ_i are unknown fit parameters. In this work, we will encounter this form twice. First in the analysis of 2-point correlation functions, where a_0 is fixed to zero while λ_1 corresponds to the ground state energy of the particle in question. The other λ_i model excited states of increasingly higher energy. While these are usually not interesting in themselves, they have to be included in the fit in order to extract a more reliable result for λ_1. Secondly, we will use a sum of exponentials ansatz in the fitting of correlated Markov chains in Section 5.3. There, the value of interest is a_0 while the exponentially decaying terms are mostly included for the sake of statistical robustness.

In Principle, any (sufficiently smooth) fitting function (or rather, the corresponding (log-)likelihood function) can be used directly in a general-purpose minimization software library, such as MINUIT[37]. But as noted by many authors (for example in [38, 39, 40]), unless the λ_i are already known, this fit is notoriously unstable. This means some additional stabilizing effort has to be spent if more than a single term is to be used reliably. In this work, we use the method of *variable projection*, first proposed by Golub and Pereyra[41]. The idea is to split the unknown variables into a "linear" and a "nonlinear" set, which in our case are the a_i and the λ_i, respectively. Then we can reformulate the maximization problem from Equation (3.24) as

$$\max_{\{\lambda_i\}} \max_{\{a_i\}} \ell(a_0, \ldots, a_n, \lambda_1, \ldots, \lambda_n)\,. \tag{3.31}$$

The trick is now that the inner maximization problem a linear fit, which for any fixed values of λ_i can be solved directly by a matrix inversion (see Section 3.2.2, without using any iterative method of uncertain convergence. The outer problem of finding the optimal λ_i is handled by a general-purpose numerical minimization routine. While theoretically equivalent to handling all variables together, in practice this splitting makes the whole process a lot more reliable. In particular, there is no need anymore to guess any starting values for the a_i.

Of course, the theoretical problem of Equation (3.30) being generally ill-conditioned still stands,[33] but any additional numerical instabilities are largely solved by this algorithm. We will use this variable projection method in two places in Chapter 5. Once for

[33] If the data is noisy, and the "true" value of multiple λ_i's lie closely together, it is impossible to distinguish their contributions. In this case, no algorithm can resolve them without using additional information such as a *prior*(see Section 3.2.1) or variational methods such as [42, 43].

the two-point correlators related to pion masses, and once for the fit to thermalization trajectories

Chapter 4

Numerical integration schemes

In this chapter, we will derive schemes for the numerical integration of various types of differential equations. After a brief overview of ordinary differential equations (ODEs) in Section 4.1, we will concentrate most effort on stochastic differential equations (SDEs). The mathematical framework of SDEs is built upon the Itô calculus, which is introduced in Section 4.2. It will turn out that this framework is not sufficient to construct higher-order schemes for SDEs in the same way as for ODEs. Therefore, in Section 4.3, we will switch the focus to studying the discretized SDE as a Markov chain, which allows us to analyze its transition operator using the Fokker-Planck equation. This finally enables us to derive higher-order integration schemes for a special kind of SDE, namely a Langevin equation. Special care is taken throughout this chapter as we are interested in the case of Lie group-valued functions, the analysis of which requires the notion of non-commuting derivatives.

4.1 Warmup: Ordinary differential equations

We start with a quick introduction to the numerical integration of ordinary (as in non-stochastic) differential equations.

4.1.1 ODEs in Euclidean space

Consider a general multivariate, autonomous, first-order differential equation

$$u : \mathbb{R}_{\geq 0} \to \mathbb{R}^n \,, \tag{4.1a}$$

$$\frac{d}{dt} u_i(t) = f_i(u(t)) \,, \tag{4.1b}$$

$$u_i(0) = u_i^{(0)} \,. \tag{4.1c}$$

We can assume that for a sufficiently smooth function f, the solution to such an ODE will usually exist uniquely, at least for a short time interval. In this work, we will not concern ourselves with any proofs of existence or uniqueness of solutions and instead only focus on numerical computation schemes, the prowess of which is ultimately not judged by theoretical analysis, but by numerical experiments.

In order to numerically compute an approximation of $u(\varepsilon)$, we can have a look at the Taylor expansion of the exact solution

$$u_i(\varepsilon) = u_i^{(0)} + \varepsilon f_i + \frac{\varepsilon^2}{2} f_{i,j} f_j + \dots \,, \tag{4.2}$$

where we introduced the shorthand notations $f_i = f_i(u^{(0)})$ and $f_{i,j} = \frac{\partial f_i}{u_j}(u^{(0)})$ and summation over the repeated index is implied.

The goal is now to construct a numerical scheme that reproduces this exact form up to some order of ε. For example, a general two-step scheme can be written as

$$u_i^{(1)} = u_i^{(0)} + k_1 \varepsilon f_i \tag{4.3a}$$

$$u_i(\varepsilon) \overset{!}{\approx} u_i^{(2)} = u_i^{(0)} + k_2 \varepsilon f_i + k_3 \varepsilon f_i^{(1)} \,, \tag{4.3b}$$

where $f^{(1)} := f(u^{(1)})$ is the force term evaluated after the auxiliary step $u^{(1)}$.

Taylor expanding this scheme gives us

$$u_i^{(2)} = u_i^{(0)} + (k_2 + k_3)\varepsilon f_i + k_1 k_3 \varepsilon^2 f_{i,j} f_j + O(\varepsilon^3) \,. \tag{4.4}$$

This now can be compared order-by-order with the expansion of the exact solution, which gives us conditions for the constants

$$\left.\begin{aligned} k_2 + k_3 &= 1 \\ k_1 k_3 &= \tfrac{1}{2} \end{aligned}\right\} \quad \Longrightarrow \quad \left\{\begin{aligned} k_2 &= 1 - \tfrac{1}{2k_1} \\ k_3 &= \tfrac{1}{2k_1} \end{aligned}\right. \,. \tag{4.5}$$

We see that the scheme is not fully fixed by these conditions and some freedom remains in choosing $k_1 \neq 0$ arbitrarily. It is easy to see that all classical schemes can thus be reproduced, such as the midpoint ($k_1 = 1/2$), Heun ($k_1 = 1$), and Ralston ($k_1 = 2/3$) methods. Right now we will not discuss how to deal with this ambiguity and in what sense one of these methods might be better than the others. We will come back to this point later when we will discover a similar ambiguity in the integration schemes for the Langevin equation.

Whichever solution we pick, our approximation of $u(\varepsilon)$ will be correct up to errors of order ε^3, which is called the *local truncation error*. When we iterate this method to compute approximations of $u(2\varepsilon), u(3\varepsilon), \dots$ up to some fixed target time t, we will need about t/ε steps to do so, each of which contributes to the overall error. Therefore the

global truncation order of the method is in $O(\varepsilon^2)$ and this is called a *second-order* scheme, which provides an improvement over the first-order *Euler scheme*

$$u_i(\varepsilon) \approx u_i^{(0)} + \varepsilon f_i \,. \tag{4.6}$$

By increasing the number of auxiliary steps in Equation (4.3), it is possible to systematically remove higher and higher orders of errors. The resulting families of schemes are called (explicit) *Runge-Kutta methods*[44, 45] which go back more than a century by now.

4.1.2 Lie group-valued ODEs

Now we generalize our notion of differential equations to a function $U : \mathbb{R}_{\geq 0} \to G$ that takes values in some Lie group G. Here we need to be careful how to even formulate the differential equation because the derivative of U at time t is an element of the tangent space $T_{U(t)}G$. So if we were to formulate the differential equation as $U'(t) = f(U(t))$ in analogy to Equation (4.1), it would be impossible to specify the function f without already knowing the solution $U(t)$.

The solution is to use the pullback of $U(t)$ to the tangent bundle to map between $T_{U(t)}G$ and the Lie algebra $\mathfrak{g} = T_{\mathbb{1}}G$. With a slight abuse of notation[34] we write the resulting differential equation as

$$U : \mathbb{R}_{\geq 0} \to G \,, \tag{4.7a}$$

$$\frac{d}{dt}U(t) = f(U(t))U(t) \,, \tag{4.7b}$$

$$U(0) = U^{(0)} \,, \tag{4.7c}$$

where we now can specify the (sufficiently smooth) function $f : G \to \mathfrak{g}$ without already knowing the solution $U(t)$.

In this setting, the first-order Euler scheme simply reads

$$U(\varepsilon) \approx \exp(\varepsilon f)\, U^{(0)} \,, \tag{4.8}$$

where, just as before, we use the shorthand $f = f(U^{(0)})$, and $\exp(\cdot) : \mathfrak{g} \to G$ denotes the exponential map.[35]

Our goal is now to derive a second-order scheme analogous to the Euclidean case. It

[34] In any concrete (matrix) representation of G, this notation works out exactly, as the pullback is just a matrix multiplication. As any computer simulation will necessarily work in a concrete representation, this notation is natural for our purposes. Furthermore, our work here is motivated by gauge theory, where all groups of interest (such as $\mathrm{SU}(N)$) are explicitly defined in terms of matrices anyway. For this reason, physicists regularly avoid talking about abstract Lie groups and only deal with concrete matrices.

[35] In any concrete representation, this is just the matrix exponential.

should read

$$U^{(1)} = \exp(k_1 \varepsilon f) U^{(0)} \tag{4.9a}$$

$$U(\varepsilon) \overset{!}{\approx} U^{(2)} = \exp\Big(k_2 \varepsilon f + k_3 \varepsilon f^{(1)}\Big) U^{(0)}\,, \tag{4.9b}$$

where $f^{(1)} = f(U^{(1)})$ as before. To fix the coefficients k_i, we could again use a Taylor expansion. But for sake of variety, let us go in a different direction. First, we re-write the differential equation as an integral equation [36]

$$U(\varepsilon) = U(0) + \int_0^\varepsilon dt\ f(U(t))U(t)\,. \tag{4.10}$$

Here, we can plug in the approximations

$$\begin{aligned} U(t) &= e^{tf} U^{(0)} + O(t^2) \\ &= (1 + tf) U^{(0)} + O(t^2)\,, \qquad (4.11a) \\ f(U(t)) &= f(e^{tf} U^{(0)}) + O(t^2) \\ &= f(U^{(0)}) + O(t) \qquad (4.11b) \end{aligned}$$

to get

$$U(\varepsilon) = U(0) + \int_0^\varepsilon dt\ \Big(f(e^{tf} U^{(0)}) + tf^2\Big) U^{(0)} + O(\varepsilon^3)\,. \tag{4.12}$$

Now we can use the well-known quadrature rule

$$\int_0^\varepsilon g(t)dt = \varepsilon g(0) + \frac{\varepsilon}{2k_1}(g(k_1\varepsilon) - g(0)) + O(\varepsilon^3)\,, \tag{4.13}$$

which works for any choice of $k_1 \neq 0$, to finally arrive at

$$\begin{aligned} U(\varepsilon) &= U^{(0)} + \varepsilon f U^{(0)} + \frac{\varepsilon}{2k_1}(f(e^{k_1\varepsilon f} U^{(0)}) + k_1 \varepsilon f^2 - f) U^{(0)} + O(\varepsilon^3) \\ &= \left(1 + \Big(1 - \frac{1}{2k_1}\Big)\varepsilon f + \frac{\varepsilon}{2k_1} f(e^{k_1\varepsilon f} U^{(0)}) + \frac{\varepsilon^2}{2} f^2\right) U^{(0)} + O(\varepsilon^3) \\ &= \exp\left(\varepsilon\Big(1 - \frac{1}{2k_1}\Big) f + \frac{1}{2k_1}\varepsilon f(e^{k_1\varepsilon f} U^{(0)})\right) U(0) + O(\varepsilon^3)\,. \end{aligned} \tag{4.14}$$

[36] As written, this is not valid for an abstract Lie group. But this computation (and all future ones) works in any arbitrary representation. So as long as the scheme itself is written in a way that makes sense in the abstract, we will just go with it.

This expression can be compared to Equation (4.9) in order to read off the conditions

$$k_2 = 1 - \frac{1}{2k_1} \qquad\qquad k_3 = \frac{1}{2k_1}\,, \tag{4.15}$$

which are exactly the same conditions as we already derived in the Euclidean case, cf., Equation (4.5). So, at least up to second order, there is no difference between integration schemes for (commutative) flat space and (generally non-commutative) Lie groups. We stress this point because it will later turn out that for stochastic differential equations, any non-commutativity will have an impact on the integration schemes. This means that introducing a stochastic term and also introducing non-commutativity is more complicated than these two extensions by themselves.

4.2 Continuous-time SDEs using Itô calculus

In this section, we will give a brief introduction to the Itô calculus[46, 47], which forms the mathematical basis of stochastic differential equations (SDE). This presentation roughly follows the book by Kloeden, Platen, and Schurz[48], though we will gloss over some of the mathematical subtleties. We will start with the definition of the Wiener process and the stochastic integral and then formulate the SDE we are aiming to numerically solve. In the derivation of a numerical integration scheme, we will see both similarities and differences compared to the non-stochastic case discussed in Section 4.1. In particular, there are two non-equivalent definitions of the truncation order of a scheme, called *strong* and *weak*.

For our ultimate goal – integrating a Langevin equation in order to sample from a known probability distribution – only weak convergence matters, which makes the full power provided by the Itô formalism unnecessary. If you are only interested in new results, you may thus skip forward to Section 4.3, in which a different approach to analyzing (weak) convergence is used to derive some novel integration scheme. For the interested reader, however, this section may serve as a connecting link between the worlds of deterministic and stochastic differential equations.

4.2.1 The Wiener process and the stochastic integral

The *standard Wiener process* $W = \{W_t \mid t \in \mathbb{R}\}$ is a stochastic process such that

1. The distribution of W_t is symmetric, i.e., $\langle W_t \rangle = 0$ for all times t.

2. The increments $W_t - W_s$ follow a Gaussian distribution with variance $|t - s|$.

3. The increments over any two non-overlapping intervals are independent, e.g.,

$$\langle (W_{t_2} - W_{t_1})(W_{t_4} - W_{t_3}) \rangle = 0 \text{ for } t_1 \leq t_2 \leq t_3 \leq t_4\,. \tag{4.16}$$

Without loss of generality, we furthermore assume that $W_0 = 0$ exactly. It can be shown that with probability one, a realization of such a process is continuous everywhere but not differentiable anywhere.

Now we can define the (Itô-) stochastic integral[46] of a function f with respect to the process W_t as a kind of Riemann integral:

$$\int_0^t f(s)\ dW_s := \lim_{\delta t \to 0} \sum_{j=1}^{n} f(t_{j-1})(W_{t_j} - W_{t_{j-1}})\,, \tag{4.17}$$

where we assume a regular partition of the interval $[0, t]$, i.e., $t_j = j\delta t$ with $\delta t = t/n$. Note that the function f can be stochastic itself but needs to be continuous, at least with probability one. In contrast to ordinary integrals, the choice of always evaluating the function f at the left boundary of the interval $[t_{j-1}, t_j]$ is important. In contrast to the Riemann-Stieltjes integral from ordinary analysis, this choice matters even after taking the limit $\delta t \to 0$.[37]

This stochastic integral shares many properties with the normal (Riemann or Lebesgue) integral such as linearity and translation invariance. However, some peculiar differences should be pointed out. First, we compute the integral of a Wiener process with respect to itself

$$\begin{aligned}\int_0^t W_s\ dW_s &= \lim_{\delta t \to 0} \sum_{i=1}^{n} W_{t_{i-1}}(W_{t_i} - W_{t_{i-1}}) \\ &= \frac{1}{2}W_t^2 - \frac{1}{2}\lim_{n\to\infty} \sum_{i=1}^{n} \left(W_{t_i} - W_{t_{i-1}}\right)^2 \\ &= \frac{1}{2}W_t^2 - \frac{1}{2}t\,,\end{aligned} \tag{4.18}$$

where the last equality holds with probability one. We see an extra term appearing when compared to the formula $\int_0^t s\ ds = t^2/2$ from ordinary analysis. This is a direct consequence of the non-symmetric definition in Equation (4.17).

Second, we will need the general formula for a change of variable. For this purpose, let

$$dx_t = \mu_t\ dt + \sigma_t\ dW_t \tag{4.19}$$

be a general combination of ordinary and Wiener measure. For an arbitrary (sufficiently smooth) variable transformation $g : \mathbb{R} \to \mathbb{R}$ and defining $\delta W_t = W_{t+\delta t} - W_t$ for a small

[37] In fact, choosing to evaluate f in the middle of the intervals leads to the *Stratonovich integral* as an alternative to Itô. While this symmetry might seem advantageous in theory, it breaks the intuitive notion of causality only flowing in the positive time direction. This means that after discretization, the process will not fulfill the Markov condition and is thus not directly suitable for the computer simulations we seek.

but finite differences we can write

$$\begin{aligned}\delta g(x_t) &:= g(x_t + \delta x_t) - g(x_t) \\ &= g'(x_t)\ \delta x_t + \frac{1}{2} g''(x_t)(\delta x_t)^2 + \dots \\ &= g'(x_t)\,(\mu_t\ \delta t + \sigma_t\ \delta W_t) + \frac{1}{2} g''(x_t)\,(\sigma_t\ \delta W_t)^2 + \dots\,, \end{aligned} \tag{4.20}$$

where we neglected all terms of order $(\delta t)^2$ or $(\delta W_t)^3$. Furthermore, δW_t is a Gaussian random variable with variance δt, so in the limit $\delta t \to 0$, the $(\delta W_t)^2$ term is equivalent to δ_t with probability one.[38] Rearranging the terms slightly and passing to the infinitesimal limit we find the *Itô formula*[46]

$$dg(x_t) = \left(\mu_t g'(x_t) + \frac{1}{2}\sigma_t^2 g''(x_t)\right) dt + \sigma_t g'(x_t)\ dW_t\,, \tag{4.21}$$

which describes a general variable transformation in the presence of stochastic terms.

Note that setting $\mu_t = 1, \sigma_t = 0$ we recover the chain rule $dg(x) = g'(x)\,dx$ from ordinary analysis. On the other hand setting $\mu_t = 0, \sigma_t = 1$, we get the special case

$$dg(W_t) = g'(W_t)\ dW_t + \frac{1}{2} g''(W_t)\ dt\,. \tag{4.22}$$

4.2.2 Euler scheme and strong convergence

A one-dimensional stochastic differential equation for an unknown stochastic process u_t is typically written as

$$du_t = a(u_t)\ dt + b(u_t)\ dW_t\,, \tag{4.23}$$

with sufficiently smooth coefficient functions a and b. For mathematical rigor, this equation should be understood as a symbolic shorthand for the integral equation

$$u_t - u_0 = \int_0^t a(u_s)\ ds + \int_0^t b(u_s)\ dW_s\,, \tag{4.24}$$

where the first term is an ordinary (Riemann or Lebesgue) integral and the second is a stochastic (Itô) integral. Just like in the non-stochastic case (cf. Equation (4.6)) we can approximate both integrands as constant to get the *Euler scheme*[39]

$$u_\varepsilon \approx u_\varepsilon^{\text{Euler}} := u_0 + a(u_0)\,\varepsilon + b(u_0)\,W_\varepsilon\,. \tag{4.25}$$

[38] As a symbolic rule, this fact is often written as $(dW_t)^2 = dt$, in analogy to $(dt)^2 = 0$.

[39] Also called *Euler-Maruyama scheme*[49] in this context.

Assessing the order of convergence of a scheme like this can be done by expanding the exact solution u_ε in powers of ε and comparing to Equation (4.25). This is more involved than in the deterministic case, however, because u is not a differentiable function. Instead, for small enough t, Equation (4.24) is dominated by the stochastic integral, which is (with probability one) of order $\sqrt{\varepsilon}$. Inserting $u_t = u_0 + b(u_0)W_t + O(\varepsilon)$ into the integral equation we get

$$\begin{aligned} u_\varepsilon &= u_0 + \int_0^\varepsilon a(u_0)\, ds + \int_0^\varepsilon b(u_0 + b(u_0)W_s)\, dW_s + O(\varepsilon^{\frac{3}{2}}) \\ &= u_0 + a(u_0)\varepsilon + \int_0^t \left(b(u_0) + b'(u_0)b(u_0)W_s\right)\, dW_s + O(\varepsilon^{\frac{3}{2}}) \\ &= u_\varepsilon^{\text{Euler}} + \frac{1}{2}b'(u_0)b(u_0)(W_\varepsilon^2 - \varepsilon) + O(\varepsilon^{\frac{3}{2}})\,, \end{aligned} \tag{4.26}$$

where we used Equation (4.18) in the last step. We see that the local truncation error of the stochastic Euler scheme is dominated by a noise term of order ε. When iterating this scheme $n \approx t/\varepsilon$ times to get an approximation of u_t for fixed t, the errors of all steps will add up. The noise terms proportional to $W_\varepsilon^2 - \varepsilon$ however are essentially independent between time steps (and all have a mean of zero), so they will partially average out. More precisely, n steps of the scheme will only increase the error proportionally to $\sqrt{n}$.

Summarizing, the stochastic Euler scheme has a *local truncation error* of order ε and a *global truncation error* of order $\sqrt{\varepsilon}$. This should be compared to the deterministic Euler scheme (cf. Section 4.1), which has local errors of order ε^2 and global errors of order ε. To be precise, we write

$$\langle |u_t - u_t^{\text{Euler}}| \rangle \in O(\varepsilon^{\frac{1}{2}})\,, \tag{4.27}$$

where the expectation value is with respect to the realization of the Wiener processes W_t. Equation (4.27) defines the so-called *strong order* of the scheme.

4.2.3 Improvement and weak convergence

The Euler scheme can be improved by including the error term of Equation (4.26) explicitly in the scheme, which is then called the *Milstein method*[50]

$$u_\varepsilon \approx u_\varepsilon^{\text{Milstein}} := u_0 + a(u_0)\,\varepsilon + b(u_0) \underbrace{\left(W_\varepsilon + \frac{1}{2}b'(u_0)(W_\varepsilon^2 - \varepsilon)\right)}_{=:\widetilde{W}_\varepsilon}\,. \tag{4.28}$$

The only difference between the (strong order $\frac{1}{2}$) Euler scheme and the (strong order 1) Milstein scheme lies in the noise term which has to be sampled from W_ε and $\widetilde{W}_\varepsilon$, respectively. To sample from $\widetilde{W}_\varepsilon$ exactly, one needs to compute the derivative of b, which often

makes this method unsuitable for high-dimensional use cases such as lattice QCD.

Alternatively, it is easy to check that $\widetilde{W}_\varepsilon$ itself is approximately Gaussian, in particular $\mathrm{Var}(\widetilde{W}_\varepsilon) = \varepsilon + O(\varepsilon^2)$. This means that the distributions of $u_\varepsilon^{\mathrm{Euler}}$ and $u_\varepsilon^{\mathrm{Milstein}}$ coincide up to corrections of order ε^2, even though their values for a particular W_ε can differ by order ε.

Iterating the scheme to a fixed time t, the distributions of u_t^{Euler}, u_t^{Milstein}, and the exact u_t are thus all equal up to corrections of order ε. To be precise,

$$|\langle g(u_t)\rangle - \langle g(u_t^{\mathrm{Euler}})\rangle| \in O(\varepsilon)\,, \tag{4.29}$$

where g is an arbitrary observable of the system and the expectation value is over all possible realizations of W_t.[40] Equation (4.29) defines the *weak order* of the scheme, which is 1 for both Euler and Milstein.

The goal of this work is to find a higher-order integration scheme for the Langevin equation which is then used to generate random samples from a given (high-dimensional) distribution. While it is possible to derive such schemes analogous to the non-stochastic Runge-Kutta methods using the Itô calculus, the size of the required computations increases rapidly. For example, just the $O(\varepsilon^{\frac{3}{2}})$ correction in Equation (4.26) consists already of four different terms proportional to the integral expressions

$$\int_0^t W_s\,ds, \quad \int_0^t W_s^2\,dW_s, \quad \int_0^t s\,dW_s, \quad \int_0^t \int_0^s W_{s'}\,dW_{s'}\,dW_s\,. \tag{4.30}$$

Any higher-order integration scheme based on this approach would need to generate random samples from these increasingly complicated expressions. This steep increase in the number of random terms for each additional power of ε (or rather $\sqrt{\varepsilon}$) is the reason very few stochastic Runge-Kutta schemes have been derived in the literature (when compared to the non-stochastic case). Additionally, in the presence of non-Gaussian noise terms (cf. Equations (4.28) and (4.30)), it is often not clear how to generalize a scheme to the multi-dimensional case. We refer to Rößler[51, 52] for some current work on this. This problem becomes especially severe in our eventual use case, where the unknown function u_t takes values in a (high-dimensional) Lie group, see Section 4.4.

Luckily, if we are only interested in weak convergence there is an alternative approach to analyzing stochastic integration schemes that will be presented in the following Section 4.3. In lattice QCD this is generally true as we are only interested in generating samples from the distribution u_t, and not in approximating any particular realization thereof.

[40] For the one-dimensional case presented here, it is convenient and sufficient to only consider the *moments* $g(x) = x^\alpha$, $\alpha = 1, 2, \ldots$. For testing convergence in high-dimensional lattice simulations, one or more physical observables should be chosen, cf. Section 4.6.

4.3 Autonomous SDEs and the transition operator

In this section, we will use an alternative approach to analyzing a stochastic differential equation. Instead of analyzing the stochastic movement of a particular realization of the process as in Section 4.2, we will look at the deterministic evolution of a probability density. This approach is especially convenient if the SDE is autonomous, i.e., the right-hand side does not explicitly depend on time. In this case, the transition operator is just an ordinary differential operator, and can thus be analyzed with more conventional methods, without any stochastic integral.

It has to be noted that this approach is strictly weaker than the one based on the Itô calculus in Section 4.2 in the sense that we can at most prove weak convergence of any numerical scheme. As our goal is the generation of random samples of a (complicated, high-dimensional) distribution, this is sufficient. In our analysis, we will go to an even weaker form of convergence, which only assures the correct stationary distribution without concern for the precise transition operator beyond first order. In principle, this is sufficient for our purposes and it does give us some extra degrees of freedom in our schemes. It is however not obvious what the full ramifications of this very lax approach are. Therefore, the numerical experiments contain both weakly convergent schemes as well as stationary-distribution convergent schemes.

4.3.1 Fokker-Planck equation for a time-discrete Markov process

Consider an autonomous, scalar, stochastic differential equation,

$$du(t) = a(u(t))\,dt + b(u(t))\,dW_t\,. \tag{4.31}$$

The Euler approximation for a time-step ε can be written as

$$u(n\varepsilon) \approx x^{(n)}\,, \tag{4.32a}$$
$$x^{(n+1)} = x^{(n)} - f(x^{(n)}, \eta^{(n)})\,, \tag{4.32b}$$
$$f(x,\eta) = -a(x)\varepsilon + \frac{b(x)}{\sqrt{2}}\eta\sqrt{\varepsilon}\,, \tag{4.32c}$$

where the $\eta^{(n)}$ are normal distributed random variables normalized as $\left\langle \eta^{(n)}, \eta^{(m)} \right\rangle = 2\delta_{nm}$. Both the normalization of η as well as the minus sign in front of f are a matter of convention, where the present choice will result in the nicest form for the transition operator in Equation (4.37). In Section 4.2, we have already discussed that this scheme does indeed recover the continuous SDE in the $\varepsilon \to 0$ limit with weak convergence order 1. In this section, we will prove that again using the transition operator.

As long as the function f does not explicitly depend on time, Equation (4.32) defines a (discrete-time) Markov process, i.e., a memoryless stochastic process. Roughly following

the steps explained in [7], we consider its *transition operator*, which gives the probability (-density) of going from one state to another in a single step. More concretely, let $P^{(n)}(x)$ be the probability distribution of $x^{(n)}$, then the probability distribution of the next step $u^{(n+1)}$ can be written as

$$P^{(n+1)}(y) = (TP^{(n)})(y) := \int dx\; T\,(y,x)\,P^{(n)}\,(x)\;, \tag{4.33}$$

where, in a slight abuse of notation, we used the letter T for both the transition operator and its integral kernel. Using a Fourier transform and a Taylor expansion, we can write this probability as

$$\begin{aligned}
T(y,x) &= \langle \delta(x-y-f(x,\eta))\rangle \\
&= \frac{1}{\sqrt{2\pi}}\left\langle \int dp \exp\left(ip(x-y-f(x,\eta))\right)\right\rangle \\
&= \frac{1}{\sqrt{2\pi}}\left\langle \int dp \exp\left(-ipf(x,\eta)\right)\right) \exp\left(ip(x-y)\right)\right\rangle \\
&= \frac{1}{\sqrt{2\pi}}\left\langle \exp\left(f(x,\eta)\partial_y\right) \int dp \exp\left(ip(x-y)\right)\right\rangle \\
&= \langle \exp\left(f(x,\eta)\partial_y\right)\rangle\, \delta(x-y) \\
&= \delta(x-y) - \langle f(x,\eta)\rangle \delta'(x-y) + \frac{1}{2}\langle f(x,\eta)^2\rangle \delta''(x-y) + \dots\;,
\end{aligned} \tag{4.34}$$

where the expectation values $\langle\cdot\rangle$ are understood with respect to η. Using integration by parts, we can write this in operator form as

$$T = \mathbb{1} + \partial\langle f(x,\eta)\rangle + \frac{1}{2!}\partial^2\langle f(x,\eta)^2\rangle + \frac{1}{3!}\partial^3\langle f(x,\eta)^3\rangle + \dots\;, \tag{4.35}$$

where the partial derivatives act on both the f's and on whatever the operator T acts on. Finally, if we replace the scalar x with a multi-dimensional variable x_i we get that the Markov process defined by

$$x_i^{(n+1)} = x_i^{(n)} - f(x_i^{(n)}, \eta^{(n)}) \tag{4.36}$$

is governed by the transition operator

$$T = \mathbb{1} + \sum_{n=1}^{\infty} \frac{1}{n!} \sum_{i_1,\dots,i_n} \partial_{i_1} \dots \partial_{i_n} \langle f_{i_1} \dots f_{i_n}\rangle\,. \tag{4.37}$$

This formula will be used in the analysis of all numerical integration schemes to come. Note that the noise $\eta^{(n)}$ can be of arbitrary dimension and follow any distribution.

4.3.2 The Langevin equation and its stationary distribution

Now we go from a general stochastic differential equation to a more specific one, namely the Langevin equation. For the specific kind of Langevin equation we are studying here, it is easy to compute (we will do so shortly) the exact stationary distribution of the process, i.e., the probability distribution of the system in the limit $t \to \infty$. Therefore we can utilize the SDE (or rather a numerical approximation thereof) in order to generate random samples of the distribution. This is particularly useful for very high-dimensional systems, where most other algorithms for generating random samples fail. Indeed, some kind of Langevin equation is at the core of essentially all modern simulations of QCD on the lattice.[41]

The Langevin equation used for generating Markov chains takes the form

$$dx_i(t) = -\frac{\partial S(x(t))}{\partial x_i} dt - \sqrt{2}\; dW_i(t)\,, \tag{4.38}$$

where S is some scalar function called the *action*, and the W_i are independent Wiener processes. The Euler approximation (see Equation (4.32)) of this SDE reads

$$x_i^{(n+1)} = x_i^{(n)} - f_i(x^{(n)}, \eta^{(n)})\,, \tag{4.39a}$$

$$f_i(x, \eta) = \varepsilon \frac{\partial S(x)}{x_i} + \sqrt{\varepsilon}\, \eta_i\,, \tag{4.39b}$$

where η is a random noise satisfying $\langle \eta_i^{(n)} \rangle = 0$ and $\langle \eta_i^{(n)} \eta_j^{(m)} \rangle = 2\delta_{nm}\delta_{ij}$.[42] Expanding Equation (4.37) for this process yields, after a short calculation, the transition operator in leading order of ε as

$$T = \mathbb{1} + \varepsilon \partial_i \left(\partial_i + \frac{\partial S}{\partial x_i} \right) + O(\varepsilon^2)\,. \tag{4.40}$$

In the limit $\varepsilon \to 0$, in which we assume to recover the continuous SDE, we get the *Fokker-Planck equation*

$$\frac{\partial P(x,t)}{\partial t} = T_{\mathrm{FP}}\, P(x,t)\,, \qquad \text{where} \quad T_{\mathrm{FP}} = \partial_i \left(\partial_i + \frac{\partial S}{\partial x_i} \right), \tag{4.41}$$

with implied summation over repeated indices. The Fokker-Planck operator T_{FP} thus describes the time evolution of a probability density function P under the continuous Langevin equation. Note that in contrast to the setting in Section 4.2 this framework does

[41] Often, it is phrased as a "hybrid Monte Carlo" algorithm, without explicitly mentioning the Langevin equation at all. But as we will see in Section 4.5.4, that algorithm is actually a discretization of the same continuous SDE.

[42] For this simple scheme, higher moments of η are irrelevant. But for higher-order schemes, it will be convenient to assume a normal distribution here.

not require any stochastic calculus but instead relies on ordinary differential operators. Assuming S to be sufficiently smooth, the time evolution will be so as well.

With a simple change of basis, the operator T_{FP} can be written as

$$T_{\mathrm{FP}} = -e^{-S/2} Q_i^\dagger Q_i e^{S/2}, \qquad \text{where} \quad Q_i = \frac{1}{2}\frac{\partial S}{\partial x_i} + \partial_i \,. \tag{4.42}$$

As this operator is obviously negative semi-definite, we know that the (continuous) Langevin equation converges to a stationary distribution $\overline{P}(x)$ which must be in the kernel of T_{FP}. This distribution can be determined by solving the differential equation

$$\begin{aligned}
0 &\overset{!}{=} T_{\mathrm{FP}}\, \overline{P}(x) && \text{(4.43a)}\\
\implies 0 &= \left(\frac{1}{2}\frac{\partial S}{\partial x_i} + \partial_i\right) e^{S(x)/2}\, \overline{P}(x) \\
&= \frac{\partial S}{\partial x_i}\, e^{S(x)/2}\, \overline{P}(x) + e^{S(x)/2}\, \frac{\partial \overline{P}(x)}{\partial x_i} && \text{(4.43b)}\\
\implies \frac{\partial \overline{P}(x)}{\partial x_i} &= -\frac{\partial S}{\partial x_i}\, \overline{P}(x) && \text{(4.43c)}\\
\implies \overline{P}(x) &= \mathrm{const} \cdot e^{-S(x)} \,, && \text{(4.43d)}
\end{aligned}$$

where the constant is in principle fixed by the normalization of the probability distribution. In this way, we have proven (up to some mathematical subtleties of course) that running the Langevin equation for long enough will always eventually result in the probability distribution e^{-S}, independent of the starting position. This means it is in principle impossible for the process to get stuck in a "wrong" region of the configuration space, though movement can be arbitrarily slow, depending on the spectral properties of $Q_i^\dagger Q_i$. Generally, it is expected in lattice field theory that there are modes in the system that slow down indefinitely when one increases the lattice volume, though few fully rigorous results are known.

The remainder of this chapter will be dedicated to deriving and evaluating higher-order numerical schemes for the Langevin equation. The goal of increasing the order of the truncation error is to increase the step size, thus moving faster through configuration space while still staying close to the correct stationary distribution.

4.3.3 Second-order schemes and convergence order

In the one-dimensional case,[43] a general two-step scheme reads

$$\widetilde{x} = x_n - \widetilde{f}\,, \qquad\qquad x_{n+1} = x_n - f\,, \tag{4.44}$$

[43] For two-step second-order schemes, the one-dimensional case leads to exactly the same coefficients as a full n-dimensional computation would, just with less clutter in the formulas. For higher orders this is no longer true.

where

$$\widetilde{f} = k_1 \varepsilon S'(x_n) + k_2 \sqrt{\varepsilon}\eta \,, \tag{4.45a}$$

$$f = k_3 \varepsilon S'(x_n) + k_4 \varepsilon S'(\widetilde{x}) + \sqrt{\varepsilon}\eta \,, \tag{4.45b}$$

and the coefficient of the very last term is already fixed to one to fix the overall scale. Using the shorthand $S' = S'(x_n)$, we can perform a Taylor expansion in $\sqrt{\varepsilon}$ to get

$$\begin{aligned} f &= \eta\sqrt{\varepsilon} + k_3 S'\varepsilon + k_4\Big(S' - \widetilde{f}S'' + \frac{1}{2}\widetilde{f}^2 S'''\Big)\varepsilon && + O(\varepsilon^{5/2}) \\ &= \eta\sqrt{\varepsilon} + (k_3+k_4)S'\varepsilon - k_2 k_4 \eta S''\varepsilon^{3/2} + \Big(\frac{1}{2}k_2^2 k_4 \eta^2 S''' - k_1 k_4 S' S''\Big)\varepsilon^2 && + O(\varepsilon^{5/2})\,. \end{aligned} \tag{4.46}$$

With this, we can compute the moments of f,

$$\langle f \rangle = (k_3+k_4)S'\varepsilon + (k_2^2 k_4 S''' - k_1 k_4 S' S'')\varepsilon^2 + O(\varepsilon^3)\,, \tag{4.47a}$$

$$\langle f^2 \rangle = 2\varepsilon + ((k_3+k_4)^2 S'^2 - 4k_2 k_4 S'')\varepsilon^2 + O(\varepsilon^3)\,, \tag{4.47b}$$

$$\langle f^3 \rangle = 6(k_3+k_4)S'\varepsilon^2 + O(\varepsilon^3)\,, \tag{4.47c}$$

$$\langle f^4 \rangle = 12\varepsilon^2 + O(\varepsilon^3)\,, \tag{4.47d}$$

and plug them into Equation (4.37) to compute the transition operator

$$T = \mathbb{1} + T^{(1)}\varepsilon + T^{(2)}\varepsilon^2 + \dots\,, \tag{4.48a}$$

$$T^{(1)} = (k_3+k_4)\partial S' + \partial^2\,, \tag{4.48b}$$

$$T^{(2)} = k_2^2 k_4 \partial S''' - k_1 k_4 \partial S' S'' + \frac{1}{2}(k_3+k_4)^2 \partial^2 S'^2 - 2k_2 k_4 \partial^2 S'' + (k_3+k_4)\partial^3 S' + \frac{1}{2}\partial^4\,. \tag{4.48c}$$

This scheme is now *weakly convergent* of order two (in the sense of Section 4.2.3) if and only if

$$T \overset{!}{=} \exp(\varepsilon T_{\mathrm{FP}}) + O(\varepsilon^3) \tag{4.49}$$

$$\iff \begin{cases} T^{(1)} \overset{!}{=} T_{\mathrm{FP}} = \partial^2 + \partial S' \\ T^{(2)} \overset{!}{=} \frac{1}{2}T_{\mathrm{FP}}^2 = \frac{1}{2}\partial^4 + \partial^3 S' - \partial^2 S'' + \frac{1}{2}\partial^2 S'^2 + \frac{1}{2}\partial S''' - \frac{1}{2}\partial S'' S' \end{cases} \tag{4.50}$$

where the exponential is the exact solution of the continuous Fokker-Planck equation (4.41). Comparing Equations (4.48) and (4.50) term by term fixes the coefficients completely to be

$$k_1 = k_2 = 1\,, \quad k_3 = k_4 = \frac{1}{2}\,, \tag{4.51}$$

which is thus the unique two-step scheme of weak order two. This scheme is analogous to the Heun method (cf. Equation (4.5)). It should be stressed however that here this scheme is unique, whereas in the non-stochastic case there was a free parameter left leading to a multitude of different schemes.

To get some freedom of choice back, we can consider an alternative definition of convergence. Any numerical scheme will produce a stationary distribution, which we write as $\overline{P}(x) = \exp(-S_{\text{eff}}(x))$ for some *effective action*

$$S_{\text{eff}}(x) = S(x) + \varepsilon S^{(1)}(x) + \varepsilon^2 S^{(2)}(x) + \dots , \tag{4.52}$$

where S is the target action and the $S^{(k)}$ are correction terms that can be computed order-by-order. In principle there are two ways to deal with these corrections:

1. In lattice QCD, the action depends on some bare parameters that have to be tuned using renormalization conditions in order to enable extrapolation to the physical continuum theory. For some combinations of actions and integration schemes, the correction terms $S^{(k)}$ do not change the universality class, i.e., the continuum limit of the action (as long as ε is not too large). Therefore, if one uses an integration scheme consistently (including when computing renormalization constants and scale settings), extrapolated results will not have an error related to the finite ε at all. We refer to the discussion in [5] for more details.

2. Tuning the coefficients of the scheme, one can make the leading correction(s) vanish, thus recovering the "correct" action to a high degree of accuracy. We say that a scheme has *stationary convergence order* k if

$$S_{\text{eff}}(x) = S(x) + O(\varepsilon^k)\,. \tag{4.53}$$

   Note that this notion of convergence is even weaker than the previously discussed "weak convergence".

Expanding the stationary equation for a k'th order scheme, we get the conditions

$$T\overline{P} = \overline{P}\,, \tag{4.54}$$

$$\begin{aligned} \implies \quad 0 &= (T - \mathbb{1})e^{-S_{\text{eff}}(x)} \\ &= (\varepsilon T^{(1)} + \varepsilon^2 T^{(2)} + \dots)e^{-S(x)} + O(\varepsilon^{k+1})\,, \end{aligned} \tag{4.55}$$

$$\implies \quad T^{(i)}e^{-S(x)} = 0 \quad \text{for} \quad i = 1, \dots, k\,. \tag{4.56}$$

We can apply these conditions (with $k = 2$) to the two-step scheme again, which yields

the conditions

$$\left.\begin{aligned} k_3 + k_4 &= 1 \\ k_2^2 &= k_1 \\ (2k_2 - k_1)k_4 &= \tfrac{1}{2} \end{aligned}\right\} \quad \Longrightarrow \quad \left\{\begin{aligned} k_3 &= 1 - \tfrac{1}{4k_2 - 2k_2^2} \\ k_1 &= k_2^2 \\ k_4 &= \tfrac{1}{4k_2 - 2k_2^2} \end{aligned}\right. \tag{4.57}$$

where k_2 is an unconstrained parameter as long as $k_2 \notin \{0, 2\}$. Setting $k_2 = 1$, we recover the weak second-order scheme from before. Any other choice yields a scheme that is only weak first-order, but still stationary second-order. A good choice is the *BPPT method*

$$k_2 = \frac{2 - \sqrt{2}}{2}, \quad k_1 = \frac{3 - 2\sqrt{2}}{2}, \quad k_3 = 0, \quad k_4 = 1\,, \tag{4.58}$$

which was derived by Torrero in [8] and demonstrated an outstandingly small finite-step-size error in their particular use case compared to other two-step schemes. Also, $k_3 = 0$ leads to a (typically minor) efficiency gain in the code.

To our knowledge, the BPPT method is the best integration scheme in use for lattice QCD today (only considering Langevin-based studies, not the more widely used HMC methods). In the next chapter, we will go beyond second order and derive a novel, even better scheme.

4.3.4 Novel third-order scheme

Using the method of stationary convergence presented in the previous section, we can derive a third-order scheme.

A general three-step scheme in arbitrary dimensions can be written as

$$x^{(1)} = x_n - f^{(1)} \qquad x^{(2)} = x_n - f^{(2)} \qquad x_{n+1} = x_n - f\,, \tag{4.59}$$

where

$$f^{(1)} = k_1 \varepsilon S'(x_n) + k_2 \sqrt{\varepsilon}\eta\,, \tag{4.60a}$$
$$f^{(2)} = k_3 \varepsilon S'(x_n) + k_4 \varepsilon S'(x^{(1)}) + k_5 \sqrt{\varepsilon}\eta\,, \tag{4.60b}$$
$$f = k_6 \varepsilon S'(x_n) + k_7 \varepsilon S'(x^{(1)}) + k_8 \varepsilon S'(x^{(2)}) + \sqrt{\varepsilon}\eta\,, \tag{4.60c}$$

and the coefficient of the very last term is already fixed to one, thus fixing the overall scale. Note that x itself can be a vector of arbitrary dimension, so is f and the derivative of f. The sheer amount of terms and plethora of indices make calculations by hand unfeasible at this point, so in this work, we can only present some key steps as well as the end result.

First, we compute a Taylor expansion of f in powers of $\sqrt{\varepsilon}$. Using the shorthands

$$S_i = \frac{\partial S(x_n)}{\partial x_i}\,, \qquad S_{ij} = \frac{\partial^2 S(x_n)}{\partial x_i \partial x_j}\,, \qquad \dots \tag{4.61}$$

for components of the derivatives of the action, the result of this (computer-aided) calculation is

$$f_i = \eta_i\sqrt{\varepsilon} + (k_6+k_7+k_8)S_i\varepsilon + A_i\varepsilon^{3/2} + B_i\varepsilon^2 + C_i\varepsilon^{5/2} + D\varepsilon^3 + O(\varepsilon^{7/2})\,, \tag{4.62a}$$

$$A_i = -(k_2k_7 + k_5k_8)S_{ij}\eta_j\,, \tag{4.62b}$$

$$B_i = -(k_1k_7 + k_3k_8 + k_4k_8)S_{ij}S_j + \frac{1}{2}(k_2^2k_7 + k_5^2k_8)S_{ijk}\eta_j\eta_k\,, \tag{4.62c}$$

$$\begin{aligned} C_i = {}& (k_1k_2k_7 + k_3k_5k_8 + k_4k_5k_8)S_{ijk}S_j\eta_k + k_2k_4k_8S_{ij}S_{jk}\eta_k \\ & - \frac{1}{6}(k_2^3k_7 + k_5^3k_8)S_{ijkl}\eta_j\eta_k\eta_l\,, \end{aligned} \tag{4.62d}$$

$$\begin{aligned} D_i = {}& \frac{1}{2}(k_1^2k_7 + k_3^2k_8 + 2k_3k_4k_8 + k_4^2k_8)S_{ijk}S_jS_k + k_1k_4k_8S_{ij}S_{jk}S_k \\ & - \frac{1}{2}(k_1k_2^2k_7 + k_3k_5^2k_8 + k_4k_5^2k_8)S_{ijkl}S_j\eta_k\eta_l - k_2k_4k_5k_8S_{ijk}S_{jl}\eta_k\eta_l \\ & - \frac{1}{2}k_2^2k_4k_8S_{ij}S_{jkl}\eta_k\eta_l + \frac{1}{24}(k_2^4k_7 + k_5^4k_8)S_{ijklm}\eta_j\eta_k\eta_l\eta_m\,. \end{aligned} \tag{4.62e}$$

From this, we calculate the expectation values $\langle f_i\rangle, \langle f_if_j\rangle, \dots, \langle f_if_jf_kf_lf_mf_n\rangle$ to plug into Equation (4.37) to get the transition operator. There is not much point in showing all the lengthy expressions here. It should just be noted that while taking the expectation values, odd powers of η vanish due to its symmetric distribution. This makes all half-integer powers of ε vanish, and we are left with a Taylor expansion

$$T = \mathbb{1} + T^{(1)}\varepsilon + T^{(2)}\varepsilon^2 + T^{(3)}\varepsilon^3 + O(\varepsilon^4)\,, \tag{4.63}$$

which we can use in the order conditions in Equation (4.54). This leads to 7 equations for the 8 unknown k_i:[44]

$$4(k_7-1)k_2k_7 + (2k_5k_8-4)k_5k_8 = (2k_2-4)k_2k_4k_8 - (4k_2-2k_5+4)k_5k_7k_8 + k_7 - 1\,, \tag{4.64a}$$

$$(2-k_2)^2k_2^2k_7 + (2-k_5)^2k_5^2k_8 = \frac{1}{3}\,, \tag{4.64b}$$

$$(2-k_2^2)k_2k_7 + (2-k_5^2)k_5k_8 = (k_2/2-k_5)k_2k_4k_8 + 7/12\,, \tag{4.64c}$$

[44] As these equations are non-linear, determining whether or not they are independent and how many degrees of freedom are left is a highly non-trivial task in general. In this case, the counting works out as one would naively expect.

$$(2-k_2)k_2k_7 + (2-k_5)k_5k_8 = \frac{1}{2}\,, \tag{4.64d}$$

$$k_1 = k_2^2\,, \tag{4.64e}$$

$$k_3 + k_4 = k_5^2\,, \tag{4.64f}$$

$$k_6 + k_7 + k_8 = 1\,. \tag{4.64g}$$

Note in particular that the bottom three conditions mean that all three steps of the scheme ($x^{(1)}, x^{(2)}$, and x_{n+1}) are themselves consistent to first order. Also similar to the two-step case (Equation (4.57)), there is exactly one degree of freedom left to choose. A simple way of fully fixing the scheme is to just set one of the coefficients to zero, thus removing the term and simplifying both any further analysis and the resulting computer code.[45] A natural choice (inspired by Equation (4.58)) is removing the second-to-last term k_7 which results in the unique[46] solution

$$\begin{aligned}
k_1 &= \frac{784}{529} - \frac{440\sqrt{3}}{529} \approx 0.041\,, & k_5 &= 1 - \frac{1}{\sqrt{3}} \approx 0.423\,,\\
k_2 &= \frac{22}{23} - \frac{10\sqrt{3}}{23} \approx 0.203\,, & k_6 &= \frac{1}{4} = 0.25\,,\\
k_3 &= \frac{31}{48} - \frac{49}{48\sqrt{3}} \approx 0.056\,, & k_7 &= 0\,,\\
k_4 &= \frac{11}{16} - \frac{47}{48\sqrt{3}} \approx 0.122\,, & k_8 &= \frac{3}{4} = 0.75\,.
\end{aligned} \tag{4.65}$$

Ultimately, of course, the merits of any numerical method must be judged by experiments in the setting where it is to be used. As it stands here, this third-order scheme is only applicable to the Euclidean Langevin equation, not to the Lie-valued case. Thus we have to delay any numerical results pertaining to lattice gauge theory until after Section 4.4, in which we will add some further terms to the scheme.

One further result of our extended experiments should be documented here. When we add a fourth step to the scheme and try to find a fourth-order scheme, the computation shows that there is no solution at all. The same is true even when we allow for multiple independent Gaussian noise terms instead of only one per dimension. This means that any potential fourth-order scheme of this kind must require (at least) 5 steps. This superlinear growth of the required number of steps for a given order is analogous to the world of non-stochastic Runge-Kutta methods. There, the well-known fourth-order scheme requires four steps, but any fifth-order scheme requires six. For us, this result is an indication that further increasing the order above three in this way is likely not going to be a worthwhile

[45] In typical lattice QCD simulations, most computing time is spent evaluating derivatives of S, which involves costly matrix inversions. Thus the computational savings of removing some k_i are usually insignificant.

[46] Disregarding some spurious solutions with coefficients outside the interval $[0, 1]$.

endeavor. As our numerical evaluation will show, the improvement from second to third order is already quite modest, so for most use cases we do not expect that going from third to fourth order will justify the increased computational cost of going from three to five steps.

4.4 Lie group valued Langevin equations

In this section, we generalize the stochastic integration schemes from Section 4.3 to the case of Lie groups. I.e., the value at any given time is not a point in $\mathbb{R}^n$, but instead an element of an arbitrary manifold admitting a (potentially non-Abelian) group structure. This is necessary to make the integration schemes applicable to gauge field theory on the lattice, where the relevant degrees of freedom take values in a gauge group, such as SU(3) in the case of QCD.

4.4.1 Properties of Lie groups and Lie derivatives

For the remainder of this chapter, let G be a compact Lie group with generators $i\tau_a$ and completely anti-symmetric structure constants f_{abc}, fulfilling the commutation relation

$$[i\tau_a, i\tau_b] = -f_{abc} i\tau_c \,. \tag{4.66}$$

Note that we use the physics convention of making factors of i explicit, which means that the τ_a are traceless, Hermitian matrices in the defining representation of SU(N). Also, the τ_a are normalized to fulfill $\mathrm{tr}(\tau_a \tau_b) = \frac{1}{2}\delta_{ab}$. This way, the indices a, b, c correspond essentially to color-indices used in QCD,[47] even though the results of this section are not specific to any particular physical theory.

Furthermore, we define the (left-) Lie derivative of any (sufficiently smooth) scalar function $g : G \to \mathbb{C}$ as

$$\partial_a g(U) = \lim_{\varepsilon \to 0} \frac{1}{\varepsilon} \left(g(e^{i\varepsilon\tau_a} U) - g(U) \right) . \tag{4.67}$$

This derivative behaves mostly the same as the usual derivative in $\mathbb{R}^N$ with respect to linearity, product rule, chain rules, Taylor series, and more. Importantly, integration by parts works as

$$\int_G dU \; g(U)(\partial_a h(U)) = -\int_G dU \; (\partial_a g(U)) h(U) \,, \tag{4.68}$$

where dU is the Haar measure of G and there are no boundary terms due to the compact-

[47] In lattice QCD, the group G is a direct sum of many copies of SU(3), one for each link of the lattice. So the indices label both the 8 colors as well as the $4V$ links.

ness of G. Note, however, that the derivative operators ∂_a, in general, do not commute, but instead fulfill the relation

$$[\partial_a, \partial_b] = -f_{abc}\partial_c \,. \tag{4.69}$$

In preparation for later calculations, we now take a closer look at the structure constants f_{abc}. In all cases we are interested in, we can choose a basis in which it is completely anti-symmetric in its three indices.[48] Furthermore, the structure constants give exactly the matrix elements of the adjoint representation which means that the (quadratic) Casimir operator in the adjoint representation can be written as

$$f_{acd}f_{bcd} = c_A\delta_{ab} \,, \tag{4.70a}$$

$$f_{ade}f_{aef}f_{afd} = \frac{1}{2}c_A f_{abc} \,. \tag{4.70b}$$

These relations are very useful when simplifying expressions involving a lot of commutators because c_A is just a constant number depending on the gauge group. In particular, in the case of SU(N), it is $c_A = N$. Whenever possible, we will prefer factors of c_A to appear in the integration scheme over explicit commutators.

Defining $\partial^2 = \partial_a\partial_a$ we can prove identities like

$$[\partial_a, \partial^2] = 0 \,, \tag{4.71a}$$

$$\partial_b\partial_a\partial_b = \Big(\partial^2 + \frac{1}{2}c_A\Big)\partial_a \,, \tag{4.71b}$$

$$\partial_b\partial_c\partial_a\partial_b\partial_c = (\partial^2 + c_A)\partial_a\partial_c\partial_a = (\partial^2 + c_A)\Big(\partial^2 + \frac{1}{2}c_A\Big)\partial_a \,. \tag{4.71c}$$

Similarly, for any function $g : G \to \mathbb{C}$ and using the shorthand notation $g_a = \partial_a g$, $g_{ab} = \partial_b\partial_a g, \ldots$, we can prove formulas for contractions such as

$$g_{ab}f_{abc} = \frac{1}{2}c_A g_c \,, \tag{4.72a}$$

$$\begin{aligned} g_{ab}g_{cd}f_{ace}f_{bde} &= -(g_{ceb} - g_{ecb})g_{cd}f_{dbe} \\ &= \frac{1}{2}c_A g_{cd}g_{cd} + g_{ecb}g_{cd}f_{dbe} \,, \end{aligned} \tag{4.72b}$$

$$g_{bc}f_{abd}f_{ace} = 0 \,, \tag{4.72c}$$

which we have implemented in a computer algebra system in order to simplify expressions like the Equations (4.62) in the non-commutative case.

[48] This is possible if and only if the group G is a direct sum of simple, compact Lie groups.

4.4.2 Integration schemes for the non-Abelian case

A general Lie group-valued Markov process can be written as

$$U_{n+1} = \exp(-i\tau_a f_a) U_n \,, \tag{4.73}$$

where $i\tau_a f_a$ is an element of the Lie algebra that depends on U_n and on some noise term(s). This is completely analogous to Equation (4.32), even though we have to write the equation multiplicatively instead of additively now in order to allow for non-Abelian groups (see the discussion at the start of Section 4.1.2 for more details). The Euler scheme for the Langevin equation now reads

$$f_a = f_a(U_n, \eta) = S_a(U_n)\varepsilon + \eta_a \sqrt{\varepsilon} \,, \tag{4.74}$$

where the η_a are independent Gaussian noise terms with variance two and $S_a = \partial_a S$ is the (Lie) derivative of the action. The analysis of this scheme works generally the same way as in the Abelian case. In particular, the transition operator of any scheme can be written as

$$T = \mathbb{1} + \sum_{n=1}^{\infty} \frac{1}{n!} \sum_{i_1,\dots,i_n} \partial_{i_1} \dots \partial_{i_n} \langle f_{i_1} \dots f_{i_n} \rangle \,, \tag{4.75}$$

which is symbolically exactly the same as Equation (4.37). The only difference is that the ∂_i operators are now non-commuting Lie derivatives. Expanding this, we find the Fokker-Planck operator

$$\langle f_a \rangle = \varepsilon \partial_a S(U) \,, \tag{4.76a}$$

$$\langle f_a f_b \rangle = 2\varepsilon \delta_{ab} + O(\varepsilon^2) \,, \tag{4.76b}$$

$$T = 1 + \varepsilon \underbrace{\partial_a (\partial_a + S_a(U))}_{=: T_{\mathrm{FP}}} + O(\varepsilon^2) \,, \tag{4.76c}$$

which, just as in the Abelian case, implies that the stationary solution is equal to e^{-S} up to $O(\varepsilon)$ corrections.

Second order

Now we can construct the non-Abelian generalization of the two-step scheme (4.44) which reads

$$\tilde{U} = \exp(-i\tau_a \tilde{f}_a) U_n \,, \qquad\qquad U_{n+1} = \exp(-i\tau_a f_a) U_n \,, \tag{4.77}$$

with

$$\tilde{f}_a = k_1 \varepsilon S_a(U_n) + k_2 \sqrt{\varepsilon} \eta_a \,, \tag{4.78a}$$

$$f_a = k_3 \varepsilon S_a(U_n) + k_4 \varepsilon S_a(\tilde{U}) + \sqrt{\varepsilon} \eta_a + k_5 c_A \varepsilon^2 S_a(U_n) + k_6 c_A \varepsilon^2 S_a(\tilde{U}) + k_7 c_A \varepsilon^{3/2} \eta_a \,. \tag{4.78b}$$

Here we have written down all possible[49] terms that might contribute to the relevant order of ε. The terms k_1 to k_4 have equivalent counterparts in Equation (4.44), the terms k_5 to k_7 are new. It turns out, however, that only one of the three new terms is relevant:

1. The k_7 term can always be absorbed into the $\sqrt{\varepsilon}$ term by rescaling the step size. Therefore we assume $k_7 = 0$.

2. The k_5 and k_6 are equivalent to leading order, so only one of them (or any single linear combination of the two) is necessary to fulfill all order conditions. In accordance with [8] we arbitrarily choose $k_5 = 0$.

3. One could consider terms with explicit commutators. The only non-vanishing one at this order would be proportional to $[\eta_b, [\eta_b, S_a]]$. After taking expectation values, one can prove (using Equation (4.70)) that this term is proportional to $c_A S_a$, thus equivalent to k_5 or k_6. Such terms will however re-appear in the three-step schemes soon.

Repeating the calculation of Section 4.3.3, we get exactly one new constraint in addition to the ones in Equations (4.57)

$$k_6 = \frac{k_2 k_4}{2} - \frac{1}{12} \,. \tag{4.79}$$

This enables us to extend any second-order scheme to the non-Abelian case, thus making it applicable to theories like lattice QCD. In particular, this is true for the previously presented BF method (4.51) (named for the authors of [5]) and BPPT method (4.58), first derived in [8].

Third order

Similarly, the novel third-order scheme (4.60) can be extended with additional non-Abelian terms to make it applicable to the simulation of non-Abelian gauge groups. Any scheme that respects gauge invariance exactly can naturally be written in terms of the Lie algebra

[49] Even though none of the quantities here have any physical dimension, a simple kind of dimensional analysis is still possible in which $[\eta] = [\sqrt{\varepsilon}]$ and $[c_A] = [\varepsilon^{-1}]$. This makes it possible to systematically list all terms up to a given order of ε.

elements

$$\eta = i\tau_a \eta_a \,, \tag{4.80a}$$

$$S'(U) = i\tau_a \partial_a S(U) \,. \tag{4.80b}$$

Using the shorthands $S_0 = S'(U_n)$, $S_1 = S'(U^{(1)})$, and $S_2 = S'(U^{(2)})$ for the three evaluations of the force term allows us to write the most general three-step scheme as

$$U^{(1)} = \exp\Big(-f^{(1)}\Big)U_n \,, \qquad U^{(2)} = \exp\Big(-f^{(2)}\Big)U_n \,, \qquad U_{n+1} = \exp\Big(-f\Big)U_n \tag{4.81}$$

with

$$f^{(1)} = \varepsilon k_1 S_0 + \sqrt{\varepsilon} k_2 \eta + \varepsilon^2 k_9 c_A S_0 + \varepsilon^{3/2} k_{10}[\eta, S_0] \,, \tag{4.82a}$$

$$f^{(2)} = \varepsilon k_3 S_0 + \varepsilon k_4 S_1 + \sqrt{\varepsilon} k_5 \eta + \varepsilon^2 k_{11} c_A S_0 + \varepsilon^{3/2} k_{12}[\eta, S_0] + \varepsilon^{3/2} k_{13}[\eta, S_1] \,, \tag{4.82b}$$

$$\begin{aligned} f = {} & \varepsilon k_6 S_0 + \varepsilon k_7 S_1 + \varepsilon k_8 S_2 + \sqrt{\varepsilon}\eta + \varepsilon^2 k_{14}[S_0, S_1] + \varepsilon^2 k_{15}[S_0, S_2] + \varepsilon^2 k_{16}[S_1, S_2] \\ & + \varepsilon^2 k_{17} c_A S_0 + \varepsilon^2 k_{18} c_A S_1 + \varepsilon^2 k_{19} c_A S_2 + \varepsilon^3 k_{20} c_A^2 S_0 \\ & + \varepsilon^{3/2} k_{21}[\eta, S_0] + \varepsilon^{3/2} k_{22}[\eta, S_1] + \varepsilon^{3/2} k_{23}[\eta, S_2] \\ & + \varepsilon^{5/2} k_{24}[\eta, [\eta, [\eta, S_0]]] + \varepsilon^{5/2} k_{25}[\eta, [\eta, [\eta, S_1]]] + \varepsilon^{5/2} k_{26}[\eta, [\eta, [\eta, S_2]]] \\ & + \varepsilon^{5/2} k_{27}[S_0, [\eta, S_1]] + \varepsilon^{5/2} k_{28}[S_0, [\eta, S_2]] + \varepsilon^{5/2} k_{29}[S_1, [\eta, S_2]] \\ & + \varepsilon^{5/2} k_{30}[S_0, [S_0, \eta]] + \varepsilon^{5/2} k_{31}[S_1, [S_1, \eta]] + \varepsilon^{5/2} k_{32}[S_2, [S_2, \eta]] \\ & + \varepsilon^{5/2} k_{33}[\eta, [S_0, S_1]] + \varepsilon^{5/2} k_{34}[\eta, [S_0, S_2]] + \varepsilon^{5/2} k_{35}[\eta, [S_1, S_2]] \,, \end{aligned} \tag{4.82c}$$

where we numbered the constants to be consistent with the Abelian case, i.e., Equations (4.60). A few remarks are in order.

1. In Equations (4.82) we included all possibilities for elements of the Lie algebra up to the desired order in ε: in $f^{(1)}$ and $f^{(2)}$ up to order ε^2, and in f up to order ε^3. Some terms that are obviously equivalent in the given order have already been omitted. Some terms in f will turn out to be equivalent, as discussed below, though this is not obvious at this stage.

2. All terms containing the quadratic Casimir operator c_A (which is just a constant number depending on the gauge group) can alternatively be written using commutators. For example, the k_9 term corresponds to $[\eta, [\eta, S_0]]$ after averaging over the noise and truncating to the order of ε we are working in. This is based on the identity $f_{acd} f_{bcd} = c_A \delta_{ab}$ of structure constants. Writing the term using c_A instead of commutators simplifies the analysis as well as the implementation of the scheme, so it is generally preferred.

3. η comes with a factor of $\sqrt{\varepsilon}$, while the S_k and c_A come with a factor of ε. The term $\sqrt{\varepsilon}\eta$ in f has no prefactor as it is used to set the overall scale.

Repeating the steps from Section 4.3.3 with this general scheme now takes considerable computational effort. In particular, the moments of f (analogous to Equations (4.62)) now contain thousands of terms in the most general case, so they are not printed here. We handled these expressions in a custom-made computer algebra system, which crucially applied all the rules from Equations (4.70) to (4.72) in order to simplify combinations of Lie derivatives. The result of this massive calculation is a system of polynomial equations for the constants k_i, which can then be solved with any general-purpose computer algebra software.

We should note that solving such a large system of polynomial equations is itself not an easy task. The performance of different computer algebra systems varies greatly and some regularly were not able to produce a solution (or prove its non-existence) within a reasonable time during our extended experimentation with different schemes. During this work, we solved the largest such systems of equations using *Maple*. At the time of writing, this is one of very few computer algebra systems that actually implements the fastest known algorithm for this purpose. [50]

The findings of our computations are as follows: After setting $k_7 = 0$, the coefficients k_1 to k_6 are also uniquely fixed (see section 4.3.4), but there remains a lot of freedom of choice in the many new non-Abelian terms. [51] Ideally, one might use this freedom to minimize the error of order ε^3 in the overall scheme. This is a rather complicated problem we defer to future work. [52] Instead, we choose to simply remove terms by setting as many of the k_i to zero as possible. First, we observe that (i) the terms k_9, k_{10}, and k_{24} do not contribute at all to the order of ε we are interested in and can thus be removed and (ii) the terms k_{28} and k_{30} are equivalent, so we choose to remove k_{28}. At this point, the solution is not yet unique. Hence, we can successively remove more and more terms and check that there still exists a solution. After removing the two explicit commutators from the intermediate step $f^{(2)}$, several of the terms containing c_A or commutators in f also drop out. The remaining space of solutions is still three-dimensional, and we have four terms with explicit commutators left, namely k_{15}, k_{21}, k_{23}, and k_{30}. Of these four, at least one has to be kept to still have a solution. We somewhat arbitrarily decided to keep k_{30}, though we tested all four possibilities in some small-scale simulations of quenched lattice QCD and were not able to see any statistically significant difference between them at all. At this point, the solution of the system of polynomial equations can be uniquely obtained

[50] First, one computes a Gröbner basis of the ideal spanned by the polynomials. Afterward, the system of equations can be solved numerically. The problem is that the Gröbner basis can contain exponentially more polynomials than the original system of equations. Therefore, advanced algorithms such as the (well-known, but rarely implemented) Faugère algorithm are necessary for its computation.

[51] In mathematical terms, the set of solutions as a subset of $\mathbb{R}^{35}$ forms a multidimensional *variety*.

[52] As there are many separate terms contributing to the ε^3 error, one would have to choose a weighting that depends on the use case, i.e., the physical lattice action to be simulated. Furthermore, for a comprehensive study, one would want to consider even more than the 35 shown in Equations (4.82), namely those that do not contribute to lower orders at all. This makes to parameter space unfeasible large.

as

$$k_{11} = \frac{127 - 73\sqrt{3}}{216} = 0.0026\,, \qquad k_{19} = \frac{7 - 3\sqrt{3}}{24} = 0.0752\,,$$
$$k_{20} = \frac{23 - 13\sqrt{3}}{288} = 0.0017\,, \qquad k_{30} = \frac{-31 + 18\sqrt{3}}{96} = 0.0018\,, \tag{4.83}$$

with k_1 to k_8 as in Equations (4.65) all other k_i vanishing. This now fully defines the novel scheme we propose in this thesis. In Section 4.6 we will numerically show that it indeed outperforms all other schemes currently used for the integration of the Langevin equation in lattice QCD.

4.5 Hamiltonian molecular dynamics

Most contemporary studies of lattice QCD that generate a statistical ensemble of gauge configurations do not explicitly employ the Langevin equation as it was discussed in the previous sections. Instead, they mostly use the *hybrid Monte Carlo* algorithm, or *HMC*, which will be explained in this chapter. The major advantage of HMC is the possibility of eliminating all effects of finite step size exactly using the *Metropolis algorithm*.

4.5.1 The Metropolis algorithm

A (time-homogeneous, time-discrete) *Markov process* is a stochastic process of generating a sequence of samples $U_1, U_2, \ldots$, in which the probability distribution of each step does only depend on its immediate predecessor. Formally, we can write this *Markov condition* (sometimes called *memorylessness*) as

$$\Pr(U_{n+1} \mid U_1, U_2, \ldots, U_n) = \Pr(U_{n+1} \mid U_n) = \Pr(U_{n'+1} \mid U_{n'}) \text{ for all } n, n' \in \mathbb{N}\,. \tag{4.84}$$

Such a process is completely characterized by its *transition operator* $T(U', U)$, which gives the probability (or probability density) of transitioning from a state U to a state U' in a single step. Given an arbitrary transition operator, it is generally a hard problem (related to the spectrum of the operator) to determine the stationary distribution of the process (assuming its existence, of course).

If however, there is a probability distribution P such that the process fulfills the *detailed balance* condition

$$P(U)\; T(U', U) = P(U')\; T(U, U') \quad \text{for all} \quad U, U'\,, \tag{4.85}$$

it is easy to see that P must be a stationary distribution of the process. The word "balance" here relates to the fact that a transition $U \to U'$ is exactly as likely to occur as the reverse, $U' \to U$. Therefore, such a process is sometimes called *reversible*.

The Metropolis algorithm[53] allows us to construct a process fulfilling detailed balance for any fixed target probability distribution P. Each step consists of two parts. First, we generate a *proposal* U' with some probability distribution $\tilde{T}$, and then we accept that proposal with probability

$$p_{\text{acc}}(U', U) = \min\left(1, \frac{P(U')\tilde{T}(U, U')}{P(U)\tilde{T}(U', U)}\right). \tag{4.86}$$

If the proposal is rejected, the previous sample U is repeated. It is a simple exercise to check that this process always fulfills detailed balance with stationary distribution P.

Note that this method works for an arbitrary proposal probability $\tilde{T}$ as long as it is guaranteed that every state has a non-zero probability to eventually reach every other state, i.e., the process is *ergodic*. In practice, however, a good choice of proposal is crucial for two reasons:

1. If $\tilde{T}$ is completely unrelated to P (such as a uniform distribution), the acceptance probability will be very low, leading to many repeated samples and thus a long thermalization and autocorrelation time.

2. If $\tilde{T}$ is too complicated, p_{acc} will be unfeasible to evaluate.

Finally, it should be noted that it is completely valid to alternate between multiple different proposal algorithms when generating the Markov chain. In this case, only the combination of them has to fulfill ergodicity. The hybrid Monte Carlo algorithm, which we discuss next, is exactly such a process composed of two different kinds of updates.

4.5.2 The hybrid Monte Carlo algorithm

Consider the task of generating a sample U with probability (proportional to) $e^{-S(U)}$ for some action S. The idea of hybrid Monte Carlo[2] (HMC) is to introduce an auxiliary variable Π (with the same number of degrees of freedom as U), following a simple Gaussian distribution. Even though this variable is statistically completely independent of U, we can consider their joint distribution

$$P(U, \Pi) \propto e^{-H(U,\Pi)} \quad \text{with} \quad H(U, \Pi) = S(U) + \frac{1}{2}\Pi^2 = V + T\,. \tag{4.87}$$

Treating Π as the conjugate momentum of U, we can use Hamiltonian dynamics to create a trajectory for these variables following the equations of motion

$$\dot{U}(t) = \Pi(t), \qquad \dot{\Pi}(t) = -S'(U(t))\,. \tag{4.88}$$

By integrating these equations for some time step Δt, we produce a new U that (in the absence of numerical errors) will exactly preserve H. Therefore, such a proposal will

always be accepted in a Metropolis algorithm (see Equation (4.86)). In order to guarantee ergodicity, it is necessary to generate new momenta Π regularly (while keeping U fixed). This is easy to do, as Π is simply Gaussian. Renewing the momenta too often, however, can have negative effects on the autocorrelation as well, cf. Section 4.5.4.

Finally, it is impossible to integrate the equations of motion without any numerical error. Therefore, H will not be exactly constant and the acceptance probability will not always be 1, but by choosing a time-reversible integration scheme, the acceptance probability (Equation (4.86)) is simplified to

$$p_{\text{acc}} = \min(1, e^{-\Delta H}), \qquad \Delta H = H(U', \Pi') - H(U, \Pi), \tag{4.89}$$

which is usually simple enough to evaluate. In the next section, we will discuss some integrators commonly used for this purpose.

4.5.3 Symplectic integrators and exponential product formulas

Formally, the solution of the equations of motion in Hamiltonian dynamics (Equation (4.88)) can always be written as

$$\begin{pmatrix} U(\Delta t) \\ \Pi(\Delta t) \end{pmatrix} = e^{\Delta t(T+V)} \begin{pmatrix} U(0) \\ \Pi(0) \end{pmatrix}, \tag{4.90}$$

where T and V are understood as linear operators acting on the combined phase space of U and Π. For most interesting cases, the operator $e^{\Delta t(T+V)}$ cannot be expressed exactly in a simple form. But we can approximate it using so-called *exponential product formulas*[54], the simplest of which is the Trotter decomposition

$$e^{\Delta t(T+V)} = e^{\Delta t T} e^{\Delta t V} + O(\Delta t^2). \tag{4.91}$$

The most basic improvement of this approximation is achieved by simple symmetrization

$$e^{\Delta t(T+V)} = e^{\frac{\Delta t}{2} T} e^{\Delta t V} e^{\frac{\Delta t}{2} T} + O(\Delta t^3). \tag{4.92}$$

In the context of Hamiltonian dynamics, this is called "leapfrog integration" because when we iterate this formula, we can imagine Π to be evaluated at discrete points $0, \Delta t, 2\Delta t, \ldots$ and U to be evaluated at $\frac{\Delta t}{2}, \frac{3\Delta t}{2}, \ldots$. Thus the two components of the system "leapfrog" over each other. The advantage of symmetry here is two-fold:

1. It implements the *symplecticity* (preservation of phase-space volumes) of the continuous Hamiltonian dynamics exactly for finite Δt. For deterministic systems, this is important for stability in long-time simulations, as it greatly dampens the energy drift that other integration schemes (such as the classical Runge-Kutta methods)

are plagued by[55, 56]. For the stochastic systems that we are interested in, this advantage is not as clear, though a positive effect on numerical accuracy might still be expected heuristically.

2. In the Metropolis algorithm, the ratio of proposal probabilities exactly cancels (Equation (4.86)), which makes evaluating the acceptance probability very simple. In fact, evaluating Equation (4.86) for the higher-order Langevin schemes from Sections 4.3.3 and 4.3.4 is completely unfeasible and would require significant work, both analytically and computationally.

Higher-order variants of Equation (4.92) have been studied for a long time (see the classical papers by Suzuki[57, 58], or [59] for a modern, more exhaustive study). In lattice QCD, the two most prominent examples are the second-order and fourth-order[53] schemes found by Omelyan et al.[3, 4], which read

$$e^{\Delta t H} = e^{\alpha \Delta t T} e^{\frac{\Delta t}{2} V} e^{(1-2\alpha)\Delta t T} e^{\frac{\Delta t}{2} V} e^{\alpha \Delta t T} + O(\Delta t^3)\,, \tag{4.93a}$$

$$e^{\Delta t H} = e^{\rho \Delta t T} e^{\lambda \Delta t V} e^{\theta \Delta t T} e^{\frac{1-2\lambda}{2} \Delta t V} e^{(1-2(\theta+\rho))\Delta t T} e^{\frac{1-2\lambda}{2} \Delta t V} e^{\theta \Delta t T} e^{\lambda \Delta t V} e^{\rho \Delta t T} + O(\Delta t^5)\,, \tag{4.93b}$$

with coefficients

$$\alpha = 0.19318\,, \tag{4.94a}$$

$$\rho = 0.1786\,, \qquad \lambda = 0.7123\,, \qquad \theta = -0.06626\,. \tag{4.94b}$$

Similar to the Langevin schemes in Sections 4.3.3 and 4.3.4, the coefficients of the scheme are not completely fixed by requiring a certain order for the overall error. Omelyan et al. fix the ambiguity by minimizing a certain norm of the leading non-vanishing order error.

4.5.4 Converting between Langevin and HMC

It can easily be checked that in the limit of small Δt, generating new random values for the momentum Π and computing the evolution $e^{\Delta t H}$ is equivalent to solving the Langevin equation with step size

$$\varepsilon = \frac{1}{2}(\Delta t)^2\,. \tag{4.95}$$

This scaling will be used in the next section in the numerical comparisons between the various Hamiltonian and Langevin schemes.

For the Hamiltonian case, there is one final complication: Usually, multiple steps of the Hamiltonian evolution are taken before generating new random momenta. This is because

[53] See Section 4.5.4 for how to relate these orders to the orders of Langevin schemes.

splitting Δt into N steps of size $\Delta t/N$ reduces the error by $1/N^2$ (for the leapfrog scheme) while splitting ε into N steps of size ε/N only reduces the overall error by $1/N$. From this perspective, it might seem advantageous to never regenerate the momenta and to just keep running the deterministic Hamiltonian dynamics. Doing so might however destroy the ergodicity of the overall process, which means some parts of the joint configuration of U and Π might be unreachable, and thus the stationary distribution of U might very subtly differ from $e^{-S(U)}$. In practice, it is common to generate new momenta at least once or twice between measurements of physical observables.

4.6 Numerical comparisons of integration schemes

For a quantitative comparison of the integration schemes, we generate gauge ensembles with many different step sizes. All of these are quenched QCD with pure Wilson gauge action (Equation (2.17)) at $\beta = 6.0$ on a 12^4 lattice with periodic boundary conditions. Our results comparing the improved Langevin schemes are shown in Figure 4.1.

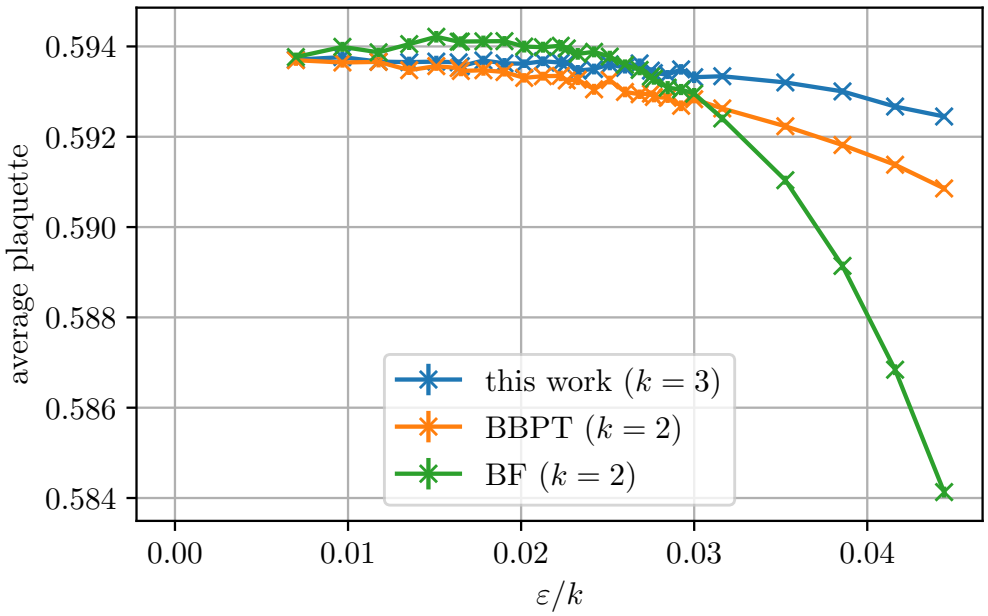

Figure 4.1: Comparison of Langevin integration schemes for quenched QCD including our novel method (Equation (4.83)). k is the number of evaluations of the force term S' per step of the integrator. Points with the same value of ε/k have the same computational cost, and thus the horizontal axis has been chosen accordingly.

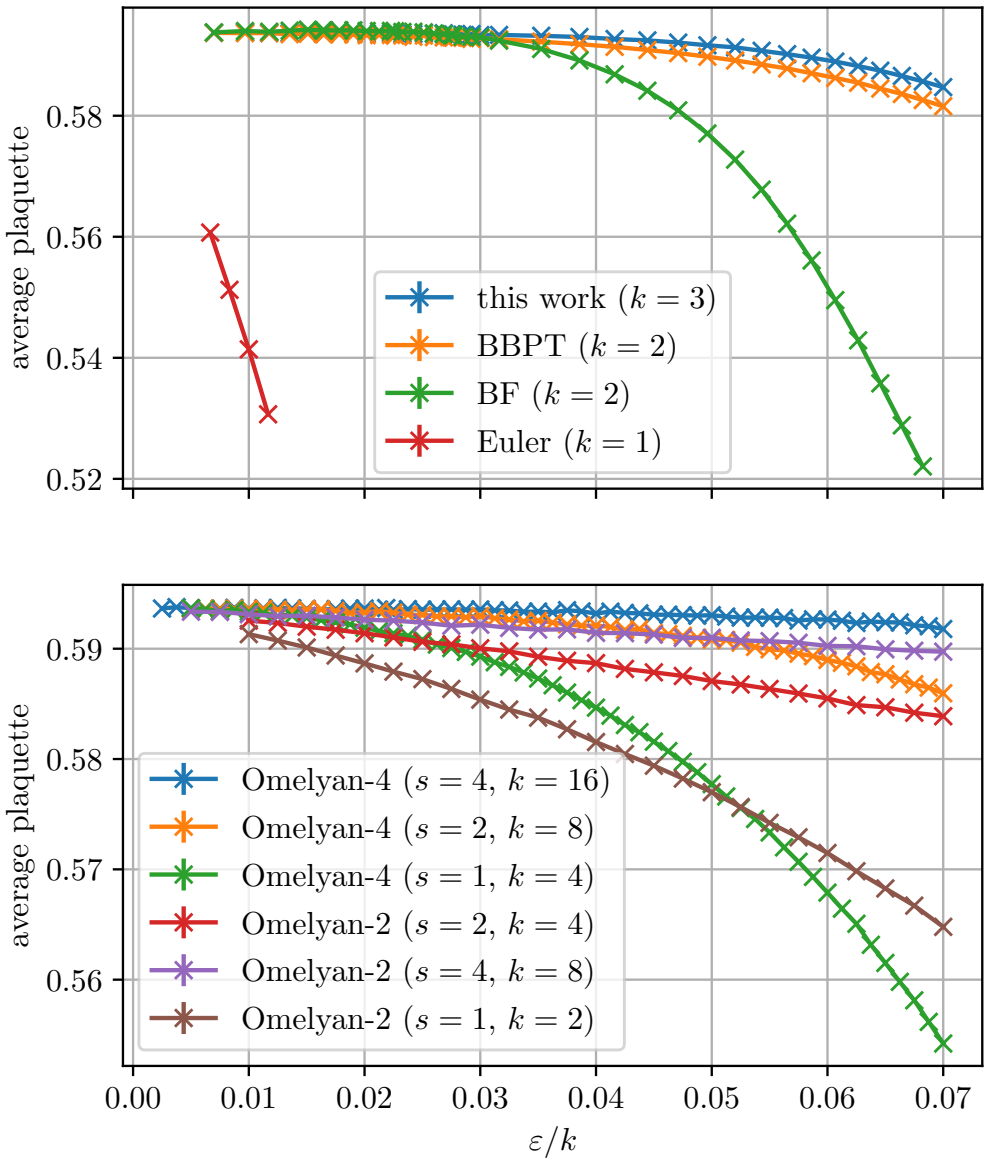

Figure 4.2: Comparison of a wide range of integration schemes for quenched QCD. k is the number of evaluations of the force term S' per step of the integrator. Top: Langevin integrators as in Figure 4.1. Bottom: Hamiltonian dynamics schemes (Equation (4.93), without accept/reject step), where $\varepsilon = (\Delta t)^2/2$ and s is the number of Hamiltonian steps taken before generating new random momenta.

In the Langevin case, one can clearly see the order-ε improvement of the BF and BPPT schemes compared to the Euler scheme. In the Hamiltonian dynamics case the different scaling behaviors are less pronounced, and the Omelyan-4 scheme with multiple substeps is the best choice here. Note that many lattice simulations implement an accept/reject

step (Metropolis algorithm), which removes all finite-ε errors exactly. In that case, a large step size will result in a low acceptance probability and thus a large autocorrelation time. Studies such as [4] suggest that the improved integrators are not worthwhile in this case and instead the Omelyan-2 scheme is already the optimal choice.

While everything works as expected in the quenched case, working with $n_f = 2$ Wilson clover fermions is more problematic due to instabilities in the Markov process. Fundamentally, these instabilities stem from so-called exceptional configurations, i.e., gauge configurations for which the fermion determinant is close to zero, and thus the Dirac operator is ill-conditioned. In practice, these points in the trajectory manifest themselves by sudden spikes of many orders of magnitude in the force term $S'(U)$. In HMC simulations with accept/reject step this is not an issue as such proposed updates are simply rejected. Without accept/reject step, such as in our case, the standard countermeasure is to adaptively adjust the step size to cancel out such spikes in the force term, see, e.g., [60]. For the purpose of this work (i.e., Chapter 5), however, we do not go that route and instead choose a simpler integration scheme and a sufficiently small step size such that the problem does not occur in the first place. In our experiments, we find that the Langevin integrators are much better behaved in this regard, though a more detailed analysis would be required to gain more insight.

Chapter 5

Correlated Markov chains for lattice QCD

In a typical lattice QCD study, the goal of generating a gauge ensemble is to have as little correlation as possible between the configurations. Therefore, the statistical error of any observable can be expected to scale as $1/\sqrt{N}$, where N is the number of gauge configurations, which is proportional to the computational effort of the study. This scaling effectively puts an upper bound on how much precision gain is possible by just increasing computational power. Thus for very high-precision studies, methodological improvements are necessary.

In this chapter, we propose a novel method for computing high-precision estimates for a class of observables that can be written as derivatives (or rather, finite differences) with respect to the bare parameters of the lattice action such as the coupling constant and the quark masses. In some applications, this method has the potential to achieve a vast decrease in statistical errors at the expense of increased systematic uncertainties by exploiting correlations between multiple Markov chains run with (very slightly) different parameters.

5.1 Idea and one-dimensional toy model

Our method consists of running two (or more) Markov chains starting from the same thermalized configuration with the same noise terms η but slightly different parameters of the action. The different chains will need a few updates to thermalize to their respective new parameters. Our method is advantageous if this thermalization can be reached before the two chains decorrelate, which they will do eventually due to the chaotic nature of the Langevin process. In the window where the two chains are thermalized and still correlated, we can compute differences of observables between the two chains with high statistical accuracy. We will demonstrate how to set up the simulation to retain correlation

for as long as possible.

To illustrate our method we first study a one-dimensional toy model. This has the advantage that we can directly follow the evolution of configurations which are now just real numbers. A simple case is the harmonic potential $S(x) = \frac{1}{2}\beta x^2$, where β represents some (inverse) coupling constant. As this potential is symmetric, the simplest non-trivial observable is the expectation value $\langle x^2 \rangle$. Analytically we have $\langle x^2 \rangle_\beta = 1/\beta$.

For any fixed value of β we can generate an ensemble of values $(x_1^{(\beta)}, x_2^{(\beta)}, \ldots, x_N^{(\beta)})$ using a suitable Langevin process. As detailed in Chapter 2, for large N the expectation value can be approximated by the average,

$$\langle x^2 \rangle_\beta \approx \frac{1}{N} \sum_{i=1}^{N} (x_i^{(\beta)})^2 \,. \tag{5.1}$$

In physics simulations, we are not only interested in particular values of the parameters but also in how the observable changes when the parameters change. For example, if β' is close to β, the difference

$$\langle x^2 \rangle_\beta - \langle x^2 \rangle_{\beta'} \approx \frac{1}{N} \sum_{i=1}^{N} \left((x_i^{(\beta)})^2 - (x_i^{(\beta')})^2 \right) \tag{5.2}$$

can be used to approximate the derivative of $\langle x^2 \rangle$ with respect to β. In the conventional approach, one generates two statistically independent Markov chains for β and β' separately, see Figure 5.1 (left). In our new approach, we generate the two chains in a way that makes them highly correlated by using the same noise terms, see Figure 5.1 (right). Note that each individual chain still follows the correct distribution determined by its action.

In Fig. 5.1 (right) we see the strong correlation between the two chains. Thus, the noise of the difference is severely suppressed. Here we used exactly the same amount of computing power for the two experiments. In this particular example, we gained about a factor of 10 in terms of noise reduction, which would roughly correspond to a factor of 100 of computing power if the same noise reduction were to be achieved purely by increasing the length of the trajectories. However, in this toy model the issues discussed in the introduction, i.e., thermalization and decorrelation of the two chains, do not play a role. We will come back to these issues below.

5.2 Results for quenched QCD

Now we turn to a physically more relevant use case for our method, namely quenched QCD, which simulates the interaction of SU(3) gauge bosons on a Euclidean lattice as introduced in Chapter 2. For this experiment, we choose the Wilson gauge action (2.17).

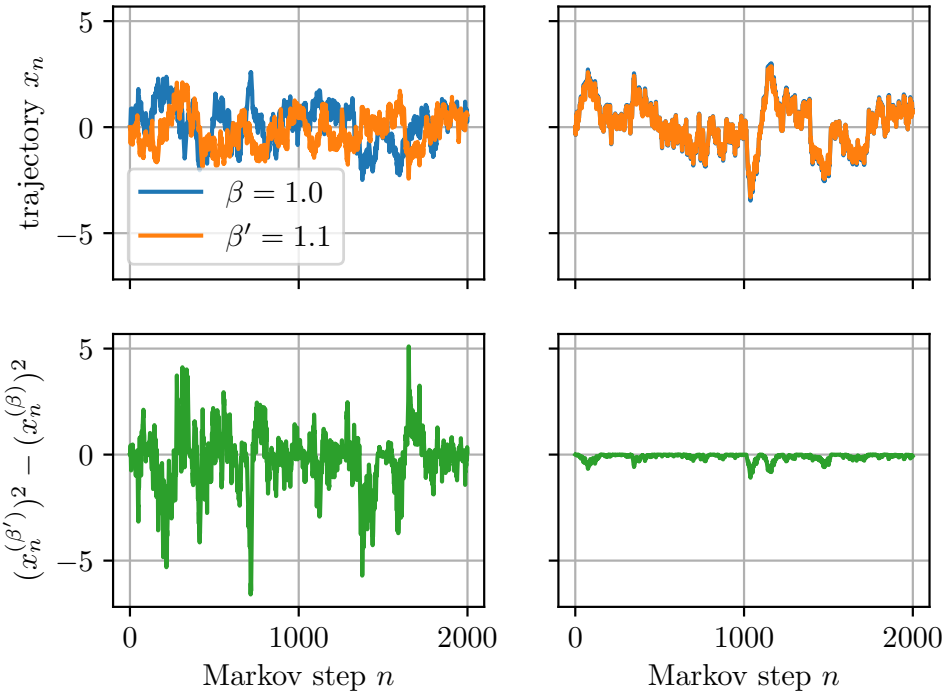

Figure 5.1: The top left plot shows two independent Markov chains for the harmonic potential at different values of the parameter β. The bottom left plot shows the difference of the quadratic observable of the two chains. The average of the difference is -0.18 ± 0.13, while the analytical value equals -0.0909. The plots on the right use correlated chains according to our new method. (In the top plot, the two curves are essentially on top of each other.) Here the average of the difference is -0.11 ± 0.01.

As an observable, we choose the average plaquette

$$\langle P \rangle = \frac{1}{18V} \sum_p \mathrm{Re}\,\mathrm{tr}\, U_p \,, \tag{5.3}$$

and consider its dependence on the parameter β around the central value $\beta_0 = 6.0$,

$$P(\beta) = C_1 + C_2(\beta - 6) + \dots \,, \tag{5.4}$$

where the derivative C_2 gets approximated by the finite difference

$$C_2 \approx C_2(\delta\beta) = \frac{P(6 + \delta\beta) - P(6 - \delta\beta)}{2\,\delta\beta} \,. \tag{5.5}$$

We proceed with our novel approach as follows: We generate a long main chain of length $T = 10^4$ (in units of Langevin time) at $\beta = 6.0$. From this, we take 50 well-spaced configurations as starting points for pairs of secondary chains of length up to $T = 20$ at

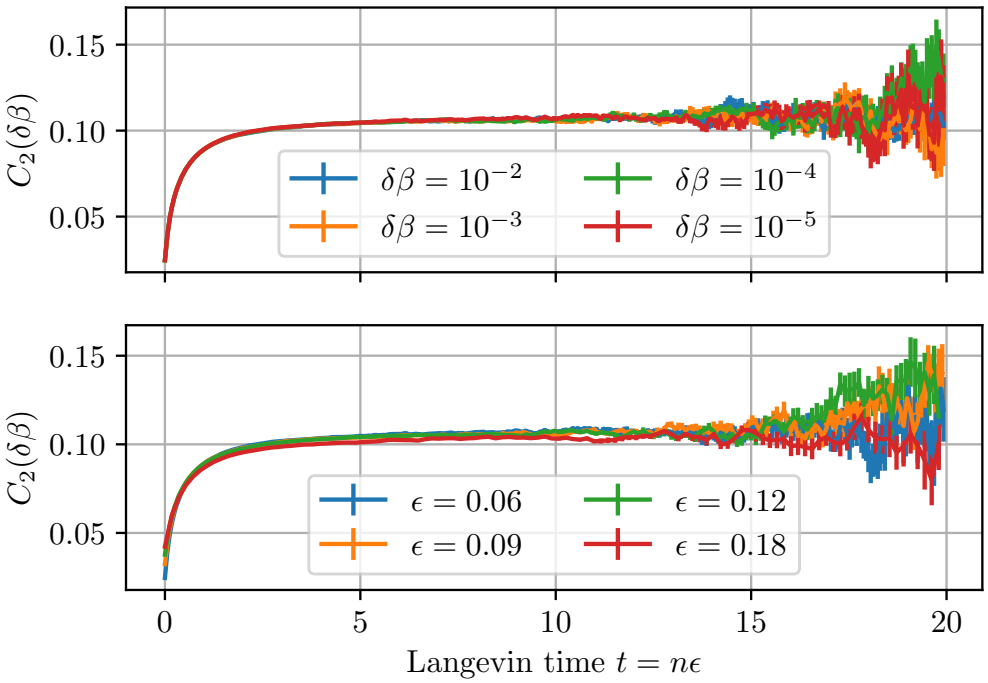

Figure 5.2: Results of our new method, running the main chain for $T = 10^4$ and each of the side chains for $T = 20$ using our novel third-order Langevin integrator. In the top panel we vary $\delta\beta$ (keeping $\varepsilon = 0.06$ fixed), while at the bottom we vary the step size ε (keeping $\delta\beta = 10^{-5}$ fixed).

$\beta = 6.0 \pm \delta\beta$. The noise terms in each pair are chosen identically, but there is no correlation between the pairs which can thus easily be used to determine statistical errors. For the integration scheme, we use our novel third-order method derived in Chapter 4. With a step size of $\varepsilon = 0.06$, this results in sufficiently small errors from the integrator, as was shown in Figure 4.2 and will again be apparent in Figure 5.2.

Figure 5.2 shows the results of such a simulation using a range of different parameters. We can see (1) the thermalization process necessary to get from $\beta = 6.0$ to $\beta = 6.0 \pm \delta\beta$ and (2) the eventual decorrelation between the two chains, visible as an (exponential) growth of statistical errors for large t. The goal is to extract results in a window between these two effects.

In Figure 5.2 (top) we observe that the statistical errors are stable over a large range of $\delta\beta$. Thus, $\delta\beta$ can be chosen small enough to make the finite difference practically the same as the exact derivative and we obtain a more precise result than would be possible with uncorrelated chains. In Figure 5.2 (bottom) we show that the eventual decorrelation of the chains does not depend on the step size ε but only on the Langevin time $t = n\varepsilon$. This indicates that the decorrelation is a feature of the continuous Langevin process itself and not an artifact of the integrator.

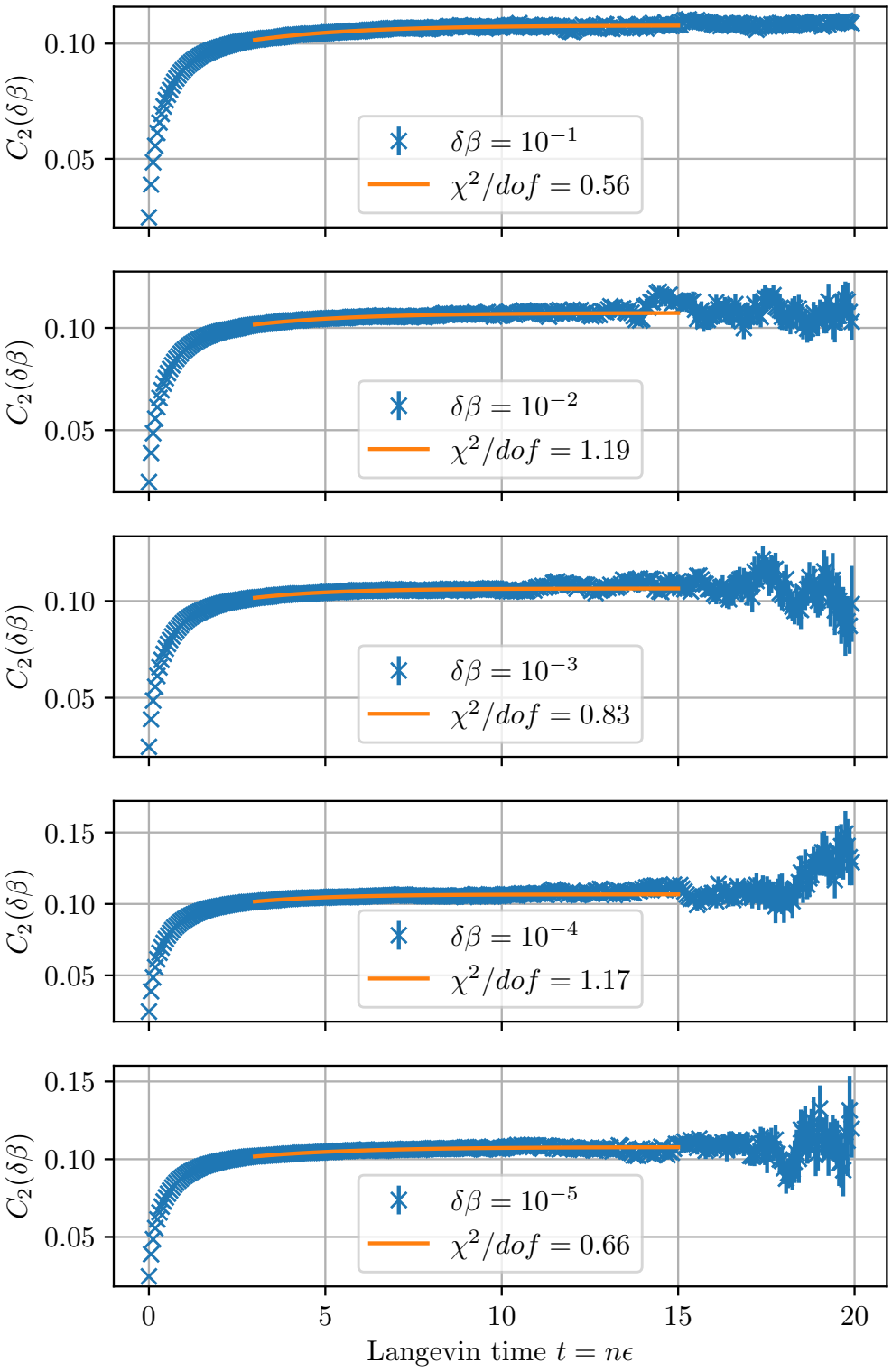

Figure 5.3: Difference between a pair of correlated chains at $\beta = 6.0 \pm \delta\beta$, starting at an identical configuration that is thermalized with $\beta = 6.0$. The integration scheme is our novel third-order Langevin integrator with step size $\varepsilon = 0.06$. The fit of the form $a_0 + a_1 e^{-\lambda t}$ in the range $t \in [3, 15]$ was done with the method discussed in Section 3.2.3.

Figure 5.3 shows the fit we use for extracting a value of $C_2(\delta\beta)$. We use 3-parameter exponential fit of the form $a_0 + a_1 e^{-\lambda t}$. Here the parameter a_0 is the eventual plateau value, while a_1 and λ explicitly parameterize the thermalization process. This way we expect more reliable results than would be possible with a simple constant fit.

Finally, we compare our results to the conventional method that uses uncorrelated Markov chains. For this, we run two independent simulations of length 10^4 at $\beta = 6.0 \pm \delta\beta$ and determine C_1 and $C_2(\delta\beta)$ from the average and the difference of the two results. Note that in this example the numerical effort of our new method ($T_{\text{total}} = 10^4 + 2 \cdot 50 \cdot 20 = 1.2 \cdot 10^4$) is lower than in the conventional method ($T_{\text{total}} = 2 \cdot 10^4$).

Results of this comparison are shown in Figure 5.4. For the old method, we observe the expected trade-off between a large systematic error – coming from approximating the derivative C_2 by the finite difference $C_2(\delta\beta)$ – for large $\delta\beta$ and large statistical errors for small $\delta\beta$. In contrast, our novel approach results in smaller statistical errors, particularly for small $\delta\beta$. Thus we can go to much smaller $\delta\beta$ than with the old method.

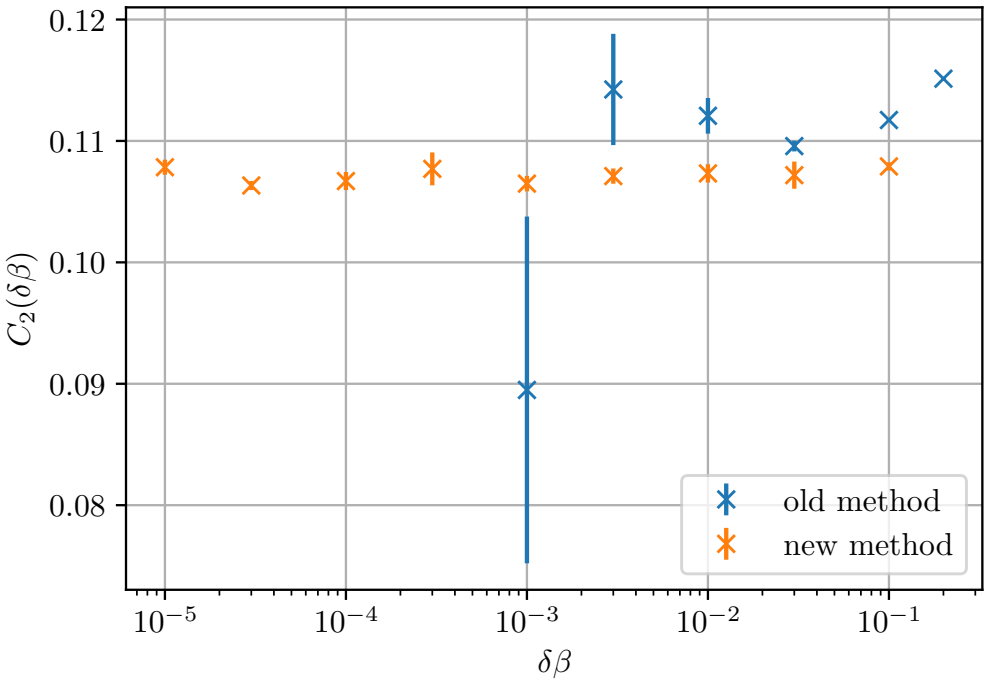

Figure 5.4: Direct comparison between the old method and our novel approach. The horizontal axis represents the finite difference used to approximate the derivative C_2. Thus in the limit $\delta\beta \to 0$ we should recover the exact derivate. The new results were obtained as shown in Figure 5.3.

As all relevant parameters, including the integration scheme, are the same between the two experiments, their results should agree within statistical errors. However, in

Figure 5.4, we see that this does not hold for the full range of $\delta\beta$. The reason might be that thermalization of the secondary chains was not fully achieved within the chosen fit range $t \leq 15$.

Therefore we repeat our analysis with a different fit. Instead of an exponential, we fit a constant in the range $t \in [15, 20]$.[54] The resulting comparison is shown in Figure 5.5. At this point, the correlation between the two chains is already diminished (as shown in Figure 5.3), thus the statistical errors increase.

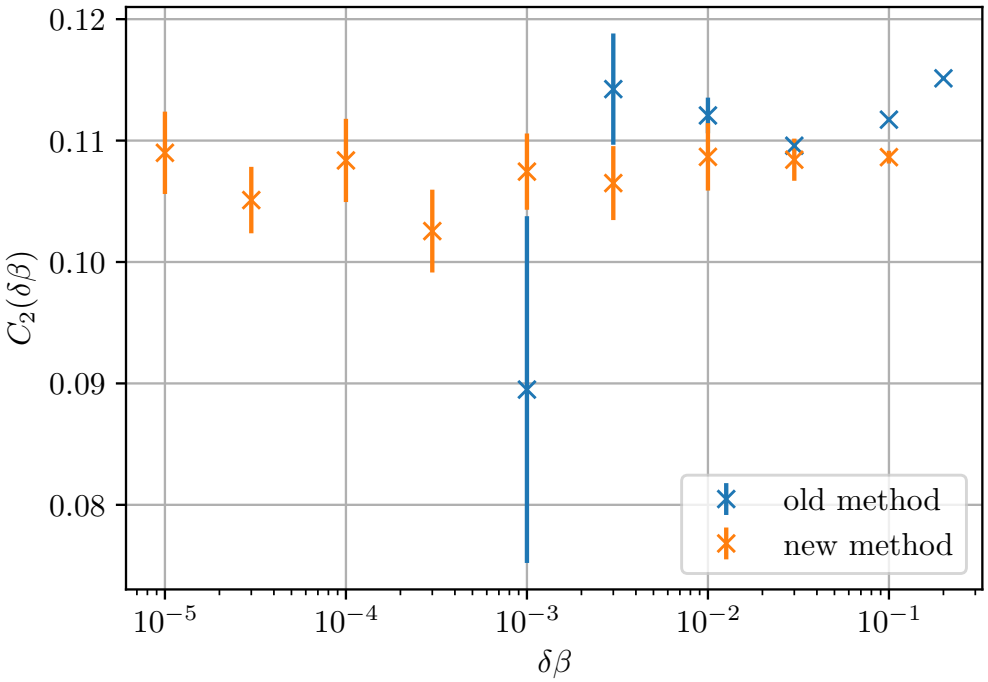

Figure 5.5: Analog to Figure 5.4. Here the new results are obtained by a constant fit over the range $t \in [15, 20]$ instead of the exponential fit over the range $t \in [3, 15]$.

Shifting the fit range even more towards larger t, one would effectively deal with uncorrelated chains again, i.e., recover the conventional method and lose any statistical benefit our method provides. Our conclusion therefore must be that – while the plots in Figure 5.2 and 5.3 look promising to the naked eye – there is no usable window between thermalization and decorrelation which would allow extraction of high-precision results in the use case studied here.

[54] In this range, statistical errors are too large to reliably fit an exponential term.

5.3 Results for QCD with $n_f = 2$ dynamical fermions

Finally in this section, we present the results of simulations with $n_f = 2$ dynamical sea quarks. We use our new method of correlated Markov chains to determine the derivative of the pion mass with respect to the bare quark mass.

Similar to the quenched case, we generate a single base ensemble with hopping parameter κ_0 and then run two chains with $\kappa = \kappa_0 \pm \delta\kappa$ using the same noise terms starting from the same configuration. Afterward, pseudoscalar correlation functions with zero momentum are measured in these side chains using the techniques outlined in Section 2.4, and pion masses are extracted by an exponential fit using the variable projection method introduced in Section 3.2.3. In contrast to the quenched case, we now use different integration schemes for the main ensemble and the side chains. The main ensemble is generated using the Omelyan-4 scheme (see Equation (4.93)) with Metropolis step (see Section 4.5.1), and the side chains are generated using our new third-order Langevin scheme with step size $\varepsilon = 0.001$. Results of such a simulation are shown in Figure 5.6.

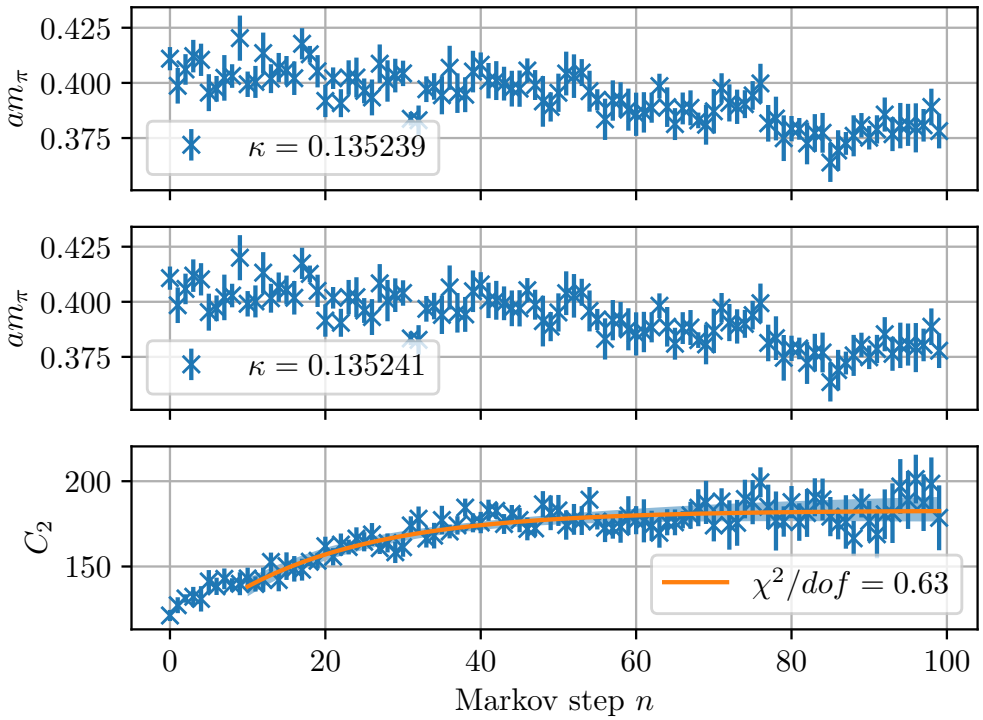

Figure 5.6: The top two plots show the pion mass obtained from the two side chains. The correlation is clearly visible in the bottom plot. The fit for the difference $C_2 = (am_\pi^{(\kappa_1)} - am_\pi^{(\kappa_2)})/(2\delta\kappa)$ is explained in the main text.

As the thermalization of the side chains takes up most of the simulation time, we fit

not only to a plateau but instead to a function of the form $a_0 + a_1 e^{-\lambda t}$ in the interval $10 \leq n \leq 100$, i.e., we explicitly parameterize leading-order thermalization effects. Just as in the quenched case, error bars are purely statistical, obtained from repeating the method for 50 statistically independent starting configurations. This way we do not need to worry about autocorrelations inside the side chains.

Note that in contrast to the plaquette observable in the quenched case, here the difference does not start at zero in the first Markov step. The reason is that we change κ from κ_0 to $\kappa_0 \pm \Delta\kappa$ for both the sea quarks (used to generate the gauge fields) and the valence quarks (used to measure the pion correlator). The former needs thermalization to take effect (about 50 Markov steps in our example), while the latter shows up immediately in the difference.

Finally, we compare the results of our new method to the conventional global-fit method. Keeping all other lattice parameters fixed, we consider the functional dependence

$$am_\pi(\kappa) = C_1 + (\kappa - \kappa_0)C_2 + \dots , \tag{5.6}$$

with $\kappa_0 = 0.13524$, $\beta = 5.2$, and lattice volume $16^3 \cdot 32$. Our simulation (see Figure 5.6) gives us a result of $C_2 = -183(14)$, whereas a conventional global fit (see Figure 5.7) gives $C_2 = -149(12)$.

For the global fit, we used 7 independent chains with 10^4 Markov steps each, while the novel approach used only one central chain at $\kappa = \kappa_0$ and $2 \cdot 50$ correlated secondary chains with 100 steps each. Thus the latter approach took less than a third as much computing time to run. While the correlations in Figure 5.6 look promising and the final result of C_2 roughly agrees with the global fit, we were not able to show a significant reduction of statistical errors.

Just as in the quenched case, the problem might lie in insufficient thermalization of the side chains. But there are some more potential sources of errors, such as exceptional configurations of the Wilson fermions (which influence the Langevin process differently than the HMC algorithm), and the particular way hadron masses are extracted from the configurations. The details thereof are relevant here because we need to estimate the mass from a single configuration in order to fully exploit the correlations, whereas, in the global-fit case, all observables are computed for a full ensemble.

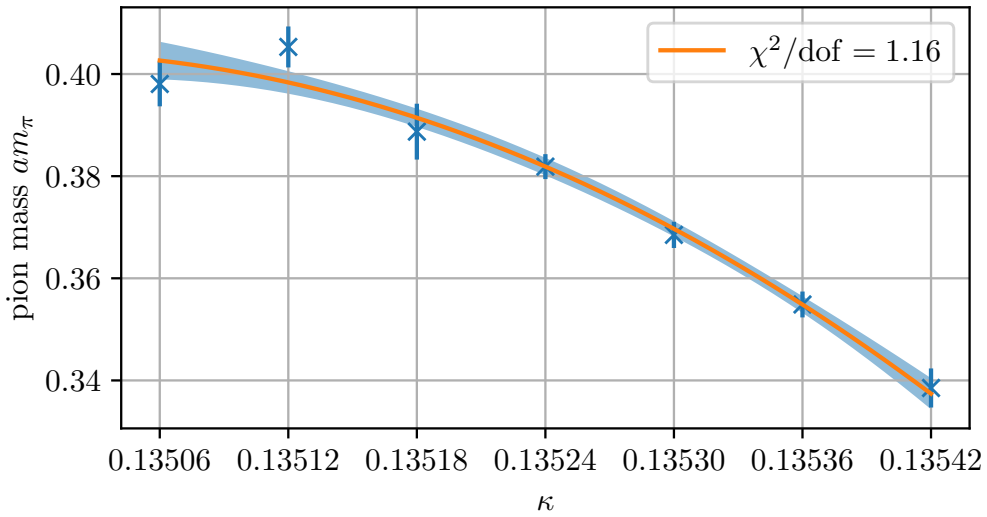

Figure 5.7: Pion mass am_π vs. hopping parameter κ. The data points are from completely independent simulations, and the fit is a quadratic polynomial. The slope at $\kappa_0 = 0.13524$ is $C_2 = -149(12)$.

Chapter 6

Summary

In this thesis, we presented the first third-order integration scheme for the Langevin equation applicable to lattice QCD. We numerically compared this method to the best previously known second-order schemes and showed a noticeable improvement. Therefore, we suggest this method for future studies of lattice gauge theory in cases where the hybrid Monte Carlo algorithm is not applicable. These cases for example include the complex Langevin method and numerical stochastic perturbation theory.

Future work might include developing a fourth-order scheme, though as our results show, returns are diminishing already at third order. Furthermore, we saw that any fourth-order scheme would require at least five evaluations of the force term, which further decreases any computational gains.

We also investigated the idea of running multiple Markov chains of a lattice simulation with the same noise terms in order to exploit the resulting correlations for high-precision measurements of the parameter dependence of physical observables. While first evaluations of this method show promising results, a more systematic comparison to the established method of global fits shows that the thermalization of the secondary Markov chains takes longer than the plots themselves suggest.

In future research, it will be worthwhile to apply the method of correlated Markov chains to other physical settings such as different lattice actions for QCD or even other field theories.

Acknowledgements

I want to thank my dad for sparking my love of science and physics a very long time ago, and also the rest of my family for encouraging me every step of the way.

I also thank my supervisor Tilo Wettig for giving me plenty of opportunities to follow my own research interests. Further thanks go to Christoph Lehner, who inspired my work on correlated Markov chains, and to Stefano Piemonte and Meinulf Göckeler who I collaborated with on research projects outside of this thesis.

I would also like to express my gratitude to my colleagues Thomas Wurm, Fabian Hutzler, Daniel Jenkins, Lorenzo Barca, Michael Gruber, Philipp Wein, and Marius Löffler. They created a humorous and thoroughly enjoyable environment both in the office and also after hours.

Last but not least, the biggest thanks go to my wife Feli. I would not have made it so far without you.

Bibliography

[1] K. G. Wilson, *Confinement of quarks*, *Physical Review D* **10** (1974) .

[2] S. Duane, A. D. Kennedy, B. J. Pendleton and D. Roweth, *Hybrid Monte Carlo*, *Physics Letters B* **195** (1987) .

[3] I. P. Omelyan, I. M. Mryglod and R. Folk, *Symplectic analytically integrable decomposition algorithms: classification, derivation, and application to molecular dynamics, quantum and celestial mechanics simulations*, *Computer Physics Communications* **151** (2003) .

[4] T. Takaishi and P. de Forcrand, *Testing and tuning symplectic integrators for the hybrid Monte Carlo algorithm in lattice QCD*, *Physical Review E* **73** (2006) .

[5] G. G. Batrouni, G. R. Katz, A. S. Kronfeld, G. P. Lepage, B. Svetitsky and K. G. Wilson, *Langevin simulations of lattice field theories*, *Physical Review D* **32** (1985) .

[6] A. Ukawa and M. Fukugita, *Langevin simulation including dynamical quark loops*, *Physical Review Letters* **55** (1985) .

[7] A. S. Kronfeld, *Dynamics of Langevin simulations*, *Progress of Theoretical Physics Supplement* **111** (1993) 293.

[8] G. S. Bali, C. Bauer, A. Pineda and C. Torrero, *Perturbative expansion of the energy of static sources at large orders in four-dimensional SU(3) gauge theory*, *Physical Review D* **87** (2013) .

[9] B. C. Hall, *Quantum theory for mathematicians*. Springer, 2013.

[10] A. Zee, *Group theory in a nutshell for physicists*. Princeton University Press, 2016.

[11] V. Bargmann and E. P. Wigner, *Group theoretical discussion of relativistic wave equations*, *Proceedings of the National Academy of Sciences* **34** (1948) .

[12] P. A. M. Dirac, *The principles of quantum mechanics*. Clarendon Press, 1981.

[13] W. Pauli, *General principles of quantum mechanics*. Springer, 1980.

[14] L. Gross, *Abstract Wiener spaces*, *Berkely Symposium on Mathematical Statistics and Probability* (1967) .

[15] C. Gattringer and C. Lang, *Quantum chromodynamics on the lattice: an introductory presentation*. Springer, 2009.

[16] H. B. Nielsen and M. Ninomiya, *Absence of neutrinos on a lattice: (I). Proof by homotopy theory*, *Nuclear Physics B* **185** (1981) .

[17] H. B. Nielsen and M. Ninomiya, *Absence of neutrinos on a lattice: (II). Intuitive topological proof*, *Nuclear Physics B* **193** (1981) .

[18] D. B. Kaplan, *A method for simulating chiral fermions on the lattice*, *Physics Letters B* **288** (1992) .

[19] P. H. Ginsparg and K. G. Wilson, *A remnant of chiral symmetry on the lattice*, *Physical Review D* **25** (1982) .

[20] H. Neuberger, *Exactly massless quarks on the lattice*, *Physics Letters B* **417** (1998) .

[21] W. Bietenholz and U.-J. Wiese, *Perfect lattice actions for quarks and gluons*, *Nuclear Physics B* **464** (1996) 319.

[22] R. Frezzotti, P. A. Grassi, S. Sint and P. Weisz, *Lattice QCD with a chirally twisted mass term*, *Journal of High Energy Physics* **08** (2001) .

[23] P. T. Matthews and A. Salam, *Propagators of quantized field*, *Il Nuovo Cimento* **2** (1955) .

[24] M. Bruno et al., *Simulation of QCD with $N_f = 2 + 1$ flavors of non-perturbatively improved Wilson fermions*, *Journal of High Energy Physics* **2** (2015) .

[25] M. Lüscher and P. Weisz, *On-Shell improved lattice gauge theories*, *Communications in Mathematical Physics* **97** (1985) [Erratum: *Commun. Math. Phys.* **98** (1985) 433].

[26] K. Symanzik, *Continuum limit and improved action in lattice theories. 1. principles and ϕ^4 theory*, *Nuclear Physics B* **226** (1983) .

[27] K. Symanzik, *Continuum limit and improved action in lattice theories. 2. O(N) non-linear sigma model in perturbation theory*, *Nuclear Physics B* **226** (1983) .

[28] M. Lüscher, S. Sint, R. Sommer and P. Weisz, *Chiral symmetry and O(a) improvement in lattice QCD*, *Nuclear Physics B* **478** (1996) .

[29] B. Sheikholeslami and R. Wohlert, *Improved continuum limit lattice action for QCD with Wilson fermions*, *Nuclear Physics B* **259** (1985) .

[30] K. Jansen and R. Sommer, *O(a) improvement of lattice QCD with two flavors of Wilson quarks*, *Nuclear Physics B* **530** (1998) .

[31] G. S. Bali, S. Collins, P. Georg, D. Jenkins, P. Korcyl, A. Schäfer, E. E. Scholz, J. Simeth, W. Söldner and S. Weishäupl, *Scale setting and the light baryon spectrum in $N_f = 2 + 1$ QCD with Wilson fermions*, *preprint* (2022) .

[32] S. Güsken, *A study of smearing techniques for hadron correlation functions*, *Nuclear Physics B - Proceedings Supplements* **17** (1990) .

[33] B. W. Bolch, *The teacher's corner: more on unbiased estimation of the standard deviation*, *The American Statistician* **22** (1968) .

[34] P. H. Garthwaite, I. T. Jolliffe and B. Jones, *Statistical inference*. Oxford University Press, 2002.

[35] S. Zacks, *The theory of statistical inference*. Wiley, 1971.

[36] R. Ruggles and H. Brodie, *An empirical approach to economic intelligence in World War II*, *Journal of the American Statistical Association* **42** (1947) .

[37] F. James and M. Roos, *Minuit: A system for function minimization and analysis of the parameter errors and correlations*, *Computer Physics Communications* **10** (1975) .

[38] W. J. Wiscombe and J. W. Evans, *Exponential-sum fitting of radiative transmission functions*, *Journal of Computational Physics* **24** (1977) .

[39] A. Ruhe and P. Å. Wedin, *Algorithms for separable nonlinear least squares problems*, *SIAM Review* **22** (1980) .

[40] C. Lanczos, *Applied analysis*. Courier Corporation, 1988.

[41] G. H. Golub and V. Pereyra, *The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate*, *SIAM Journal on Numerical Analysis* **10** (1973) .

[42] C. Michael, *Adjoint sources in lattice gauge theory*, *Nuclear Physics B* **259** (1985) .

[43] M. Lüscher and U. Wolff, *How to calculate the elastic scattering matrix in two-dimensional quantum field theories by numerical simulation*, *Nuclear Physics B* **339** (1990) .

[44] C. Runge, *Über die numerische Auflösung von Differentialgleichungen*, *Mathematische Annalen* **46** (1895) .

[45] W. Kutta, *Über die numerische Auflösung von Differentialgleichungen*, *Zeitschrift für Mathematik und Physik* **46** (1901) .

[46] K. Itô, *Stochastic integral*, *Proceedings of the Imperial Academy* **20** (1944) .

[47] K. Itô, *On stochastic differential equations*. American Mathematical Society, 1951.

[48] P. E. Kloeden, E. Platen and H. Schurz, *Numerical solution of SDE through computer experiments*. Springer, 2012.

[49] G. Maruyama, *Continuous Markov processes and stochastic equations*, *Rendiconti del Circolo Matematico di Palermo* **4** (1955) .

[50] G. N. Mil'shtejn, *Approximate integration of stochastic differential equations*, *Theory of Probability & Its Applications* **19** (1975) .

[51] A. Rößler, *Second-order Runge–Kutta methods for Itô stochastic differential equations*, *SIAM Journal on Numerical Analysis* **47** (2009) .

[52] A. Rößler, *Runge–Kutta methods for the strong approximation of solutions of stochastic differential equations*, *SIAM Journal on Numerical Analysis* **48** (2010) .

[53] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, *Equation of state calculations by fast computing machines*, *Journal of Chemical Physics* **21** (1953) .

[54] H. F. Trotter, *On the product of semi-groups of operators*, *Proceedings of the American Mathematical Society* **10** (1959) .

[55] K. D. Hammonds and D. M. Heyes, *Shadow Hamiltonian in classical NVE molecular dynamics simulations: A path to long time stability*, *Journal of Chemical Physics* **152** (2020) .

[56] K. D. Hammonds and D. M. Heyes, *Shadow Hamiltonian in classical NVE molecular dynamics simulations involving Coulomb interactions*, *Journal of Chemical Physics* **154** (2021) .

[57] M. Suzuki, *Generalized Trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems*, *Communications in Mathematical Physics* **51** (1976) .

[58] M. Suzuki, *On the convergence of exponential operators—the Zassenhaus formula, BCH formula and systematic approximants*, *Communications in Mathematical Physics* **57** (1977) .

[59] N. Hatano and M. Suzuki, *Finding exponential product formulas of higher orders*, *Lecture Notes in Physics* **679** (2005) .

[60] G. Aarts, F. A. James, E. Seiler and I. Stamatescu, *Adaptive stepsize and instabilities in complex Langevin dynamics*, *Physics Letters B* **687** (2010) .