A Study of
Improved Monte-Carlo Methods
for Lattice Gauge Theories

Michael James Peardon

Doctor of Philosophy

The University of Edinburgh

1995

To Mum and Dad.

Abstract

This thesis is concerned with the study and improvement of methods for generating Monte-Carlo configurations used for providing non-perturbative numerical results from lattice gauge theories such as QCD, the theory of strong interactions between quarks and gluons.

At present, lattice calculations require large amounts of CPU time on the largest supercomputers. In spite of this numerical assault, the majority of results generated still contain systematic errors from the use of the quenched approximation. In this approximation, employed to dramatically reduce computational costs, the effects of quantum fluctuations in the vacuum of fermion fields are ignored.

Chapter 2 investigates the efficiency of a new approximate technique for dynamical fermion simulations which replaces the fermion action with the action of a large number of flavours of locally interacting auxiliary boson fields. The technique is shown to have problematic behaviour in the approach to the limit in which it exactly reproduces the required lattice gauge theory. The autocorrelation time, a measure of efficiency is shown to rise linearly in the number of boson fields employed.

Chapter 3 proposes an improvement to this developing method which removes the bias of the approximation introduced. This avoids the computationally difficult approach to the exact limit of the approximation.

Chapter 4 involves the calculation of the mass of the scalar glueball of QCD using large lattice spacings to avoid the high penalty for the approach to the continuum limit with an "improved" lattice action to remove the significant discretisation artifacts present at these spacings.

Declaration

This thesis was composed by me and contains my own work, carried out as a member of the UKQCD collaboration. The work discussed in chapter 4 was done in collaboration with Colin Morningstar.

Results contained in chapters 2 and 3 have been published in

- Nucl. Phys. B (Proc Suppl.) **42**, 891 (1995) `hep-lat/9412008`.

Results from chapter 4 will appear in

- C. Morningstar and M. Peardon.

Proceedings of Lattice '95 (Melbourne) `hep-lat/9509069`.

Mike Peardon

Acknowledgements

I would like to thank my supervisors, Ken Bowler and Brian Pendleton for their support and allowing me the freedom to explore the subject during my three years in Edinburgh. I am grateful to my collaborator during the last year of my study, Colin Morningstar. I would also like to thank Dave Henty for his unbelievable patience; I am indebted to him for his help and encouragement during the first two years of my PhD. I am grateful to PPARC for financial support during my studies and for providing a travel grant to Vienna and to the 015 grant for providing far too many travel grants after I had messed up applications elsewhere.

- "Can I just say 'Hello' to my Mum, my Dad and everyone that knows me?"
- "I think you just have!"

Mike.

Contents

Chapter 1

Introduction

In Chapter 1, an introduction to aspects of configuration generation on computers, which is the foundation for current Monte-Carlo methods in lattice gauge theory, is presented. A more detailed introduction to the lattice formulation of a quantum field theory can be found in *eg.* [1, 2].

1.1 Quantum Field Theory on the Lattice

The lattice provides a method for making non-perturbative, theoretical predictions directly from the path-integral formulation of field theories. It also introduces a natural cut-off to regulate the theory.

To formulate a lattice field theory, Euclidean space time is discretised into a regular hypercubic grid of points, each point separated from its neighbour by the lattice spacing, a. Results for continuum physics should be reproduced as the lattice spacing is taken to zero. The renormalisation group prediction is that, in this regime, the ratio of physical quantities should become independent of the lattice discretisation and also that, in lattice units, the correlation length of the lightest mode of the lattice system must diverge.

With the theory mapped onto a finite-volume lattice, non-perturbative calculations become accessible to computational methods. The approach to the continuum, however, leads to a high cost since the infinite correlation length implies it is the critical point of a statistical mechanics system. This means efficient simulation techniques, both in the choice of lattice discretisation scheme and numerical algorithms, should be sought.

Since this thesis relates to the study of gauge theories, where gauge bosons interact with fermions via a gauge-invariant action, the discussion of this chapter will be restricted to the computational study of these theories on the lattice.

1.1.1 Lattice Gauge Fields

The gauge boson fields in the continuum define a gauge covariant parallel transport along a path which is a member of the gauge group.

$$U = Pexp^{ig\int_x^y A_\mu(s)ds_\mu}. \tag{1.1}$$

Under a local gauge transformation, $g(x)$,

$$U \rightarrow g(x)\; U\; g^\dagger(y). \tag{1.2}$$

This provides a method of discretising the gauge fields; a gauge variable, existing on the link between adjacent lattice points (x and $x+\hat{\mu}$) is defined with the gauge covariance of (1.2) and with the integral of (1.1) replaced by a mid-point approximation.

$$U_\mu(x) = exp(iag\; T^a A^a_\mu(x+\frac{\hat{\mu}}{2})) \in SU(N). \tag{1.3}$$

A^a_μ are the gauge boson fields and T^a are the generators of the fundamental representation of $SU(N)$. For simulations on a finite volume lattice (length L), this discretisation leads to periodic boundary conditions for the boson fields if the lattice fields $U_\mu(x)$ and $U_\mu(x+\hat{\nu}L)$ are identified.

The trace of a path-ordered product of these link variables around a closed loop is a gauge invariant quantity, called a Wilson loop. The smallest such non-trivial path is around a 1×1 square called the plaquette.

$$U_\Box(x) = U_\mu(x)U_\nu(x+\hat{\mu})U^\dagger_\mu(x+\hat{\nu})U^\dagger_\nu(x). \tag{1.4}$$

The fermion fields can be directly related to their lattice counterparts and exist on the sites themselves. On performing a gauge transformation, they become

$$\psi(x) \rightarrow g(x)\psi(x). \tag{1.5}$$

A gauge invariant fermion bilinear can be built on the lattice by connecting fermion fields on different sites with a path-ordered product of gauge link variables. The simplest such bilinear involving fields on different sites is

$$\Delta = \bar{\psi}(x)U_\mu(x)\psi(x+\hat{\mu}). \tag{1.6}$$

1.1.2 The Lattice Gauge Action

Having discretised the continuum fields into their lattice variables, consider now how the theory of interest is simulated. The theory is defined via its action and a lattice counterpart must be constructed. In the continuum, the gauge action is

$$S_g = \frac{1}{4}\int d^4x \,\mathrm{Tr}\, F_{\mu\nu}(x)F_{\mu\nu}(x). \tag{1.7}$$

Any choice of discretisation is not unique; there exist an infinite number of lattice actions which reproduce the continuum action in the limit $a \to 0$. The simplest lattice discretisation, the Wilson action is discussed here. An action involving more lattice operators, chosen to reduce the discretisation artefacts is discussed and tested in chapter 4.

Wilson [3] proposed a simple lattice action of the form

$$S_G = -\beta \sum_x \sum_{\mu>\nu} \mathrm{ReTr}\; U_\Box(x), \tag{1.8}$$

where the lattice coupling, β for an $SU(N)$ gauge theory is related to the bare gauge coupling, g by

$$\beta = \frac{2N}{g^2}. \tag{1.9}$$

Simulating with this action and a finite lattice spacing introduces $O(a^2)$ discretisation errors so any quantity computed on the lattice will have errors at least of this order.

1.1.3 The Lattice Fermion Action

A naive discretisation of the continuum action for fermions of bare mass m, coupled to the gauge bosons,

$$S_f = \int d^4x\, \bar{\psi}(x)(\not{D} + m)\psi(x), \tag{1.10}$$

leads to the famous fermion doubling problem. Here, the massless free field propagator has 2^d poles in the Brillouin zone corresponding to 2^d flavours of fermions. Nielsen and Ninomiya [4] showed that the fermion doubling problem is a fundamental one for a lattice discretisation of the fermion action; one fermion flavour can not be simulated on the lattice using a local, Euclidean invariant, and translation independent action without breaking chiral symmetry.

The most commonly used method for removing the unwanted flavours for QCD calculations of hadron masses and matrix elements is the Wilson fermion matrix [5] and its Symanzik-improved counterpart, the Sheikholeslami-Wohlert [6] fermion matrix. In chapters 2 and 3, where dynamical fermion simulations are discussed, the staggered fermion formulation of Kogut and Susskind [7] is considered. Here, a spatially-distributed representation of the Dirac matrices is employed and diagonalised to reduce the number of fermion flavours from 2^d to $2^{\frac{d}{2}}$. For staggered fermions, the fields on a single site have only colour (gauge symmetry) indices. The spin and flavour structure of the system is distributed over the sites in a 2^d hypercube. The staggered fermion $\not{D}$ preserves a $U(1) \times U(1)$ remnant of the full chiral symmetry which is explicitly broken by the additional term added to the naive fermion matrix to remove the doublers in the Wilson scheme. This means that the bare mass of Wilson fermions must be calculated by recovering the broken chiral symmetry. This is achieved by tuning the Wilson "hopping parameter", κ until the pion (the Goldstone boson) mass vanishes. This makes staggered fermions easier for testing Monte-Carlo dynamical fermion algorithms since a direct choice of bare fermion mass can be made.

The matrix for staggered fermions of bare mass m, coupled to the gauge boson field in a gauge-invariant, manner is

$$M_{x,y}[U] = m\delta_{x,y} + \frac{1}{2}\sum_{\mu} \eta_\mu(x) \left(\delta_{x+\mu,y} U_\mu(x) - \delta_{x-\mu,y} U_\mu^\dagger(x-\mu) \right) \tag{1.11}$$

with $\eta_\mu(x)$ the staggered fermion phases giving the Dirac structure in the continuum.

$$\eta_\mu(x) = (-1)^{x_1+x_2+\ldots+x_{\mu-1}}, \eta_1(x) = 1. \tag{1.12}$$

Then the lattice fermion action is

$$S_F = \sum_{x,y} \bar{\chi}(x) M_{x,y}[U] \chi(y). \tag{1.13}$$

Combining the gauge and fermion actions together gives the full lattice partition function

$$Z = \int \mathcal{D}U \mathcal{D}\chi \mathcal{D}\bar{\chi}\, e^{-S_G + \bar{\chi} M \chi}. \tag{1.14}$$

$\mathcal{D}U$ is a gauge invariant measure on configuration space.

$$\mathcal{D}U = \prod_{x,\mu} dU_\mu(x). \tag{1.15}$$

For QCD, this measure is built from the Haar measure on the $SU(3)$ group manifold. For $U(1)$ gauge theories, the exponential map between the group and its Lie algebra is simple (for a link variable, $U_\mu(x) = exp(iag\ A_\mu(x)) \in U(1)$) and the measure can be written explicitly.

$$\mathcal{D}U_{U(1)} = \prod_{x,\mu} d\theta_\mu(x). \tag{1.16}$$

Grassmann-valued variables can not be simulated stochastically on computers. To proceed, the bilinear fermion action is integrated analytically to give an effective action, dependent on the gauge fields alone.

$$Z = \int \mathcal{D}U \det M[U]\ e^{-S_G[U]}. \tag{1.17}$$

This is responsible for the high cost of lattice gauge simulations using this action. The fermion matrix determinant directly couples every gauge link to every other link on the entire lattice. This means that updating a single link variable requires an extremely intensive lattice-wide calculation of $\det M$. The majority of simulations of QCD are carried out in the quenched approximation. Here, the fermion determinant is not simulated as part of the action and the pure gauge theory of QCD with gauge boson fields interacting via the local Wilson action only is simulated.

$$Z = \int \mathcal{D}U \; e^{-S_G[U]}. \tag{1.18}$$

Fermions are included only as external state propagators. Quenching the theory implies that in the quantum vacuum the fermion fields do not fluctuate. This corresponds to all Feynman graphs with internal fermion loops being ignored. The success of the quenched approximation for QCD is due to the dominance of gluon dynamics (as indicated by the success of many potential models for heavy quark systems) and the possible inclusion of quark loop effects into a renormalisation of the coupling constant. The quenched approximation is, however, an important systematic error in simulations.

1.1.4 The Transfer Matrix

As mentioned in the opening to section 1.1, the Euclidean lattice discretisation of a quantum field theory maps the theory onto a statistical mechanics system.

The formalism of the transfer matrix provides a direct link between the quantum field theory and the statistical mechanics problem defined on the lattice. The transfer matrix is a linear operator on the Hilbert space of quantum states. It is related to the quantum Hamiltonian of the system by

$$T = \lim_{a \to 0} e^{-aH} \tag{1.19}$$

and thus acts as a time evolution operator for the system, mapping states at time t to states at time $t + a$. The partition function of the statistical mechanical system on a lattice of temporal extent N, is then related to T by

$$Z = \mathrm{Tr}\; T^N. \tag{1.20}$$

The trace is over the Hilbert space of states.

The link between the lattice and quantum mechanics allows a lattice definition of a quantum mechanical operator, and thus the spectrum of gauge field theories, to be made. With the Wilson gauge action, timeslice t is linked only to timeslices $t \pm 1$. A complete set of states can be defined on each timeslice, with the transfer matrix linking states on neighbouring timeslices. The transfer matrix is thus a hermitian, positive definite operator with eigenvalues related to the mass spectrum of the theory. Analysis of this set of quantum states then allows predictions about the long-time behaviour of correlators of lattice operators to be made. In particular, the effective mass of a physical particle m_{eff}, extracted by examining the exponential decay of a correlator built from operators, $\mathcal{O}(t)$ with the appropriate quantum numbers of the particle

$$m_{\text{eff}}(t) = -\ln\left[\frac{\langle\mathcal{O}(0)\mathcal{O}(t)\rangle}{\langle\mathcal{O}(0)\mathcal{O}(t+1)\rangle}\right], \tag{1.21}$$

can be shown to converge to its asymptotic form from above.

In chapter 4 problems arising when defining the transfer matrix for an improved lattice action are discussed. These occur when a term is added to the Wilson gauge action which links sites separated temporally by two lattice spacings. Now, the analysis mentioned above, which relies on being able to define a complete set of quantum states on a single timeslice breaks down. This has important consequences for masses extracted from correlator decays.

1.2 Improving Lattice Actions

The Wilson action of (1.8) can be shown to be a discretisation of the continuum gauge kinetic terms with $O(a^2)$ errors. The popular Wilson fermion action used in the majority of hadronic calculations in QCD has $O(a)$ discretisation errors. Both of these discretisation schemes thus introduce a maximum lattice spacing beyond which the lattice artefacts introduced become too large to allow the extrapolation to the continuum to be reliably carried out.

1.2.1 Symanzik Improvement

Symanzik introduced a scheme whereby the discretisation errors in a particular operator can be removed systematically order-by-order by the addition of extra higher-dimensional (and thus trivial) operators. To improve a dimension n operator with $O(a^m)$ discretisation errors requires the use of dimension $n+m$ operators with the appropriate quantum numbers

$$\tilde{X} = X^{(n)} + \sum_i \alpha_i X_i^{(n+m)}. \tag{1.22}$$

For a local operator, the coefficients of each term in the improvement scheme can be calculated reliably in perturbation theory.

If the improvement scheme is to correctly remove lattice artefacts, then it must be applied to the lattice action, to remove "dynamical" cutoff effects as well as to the operator of interest.

The scheme allows for improvement to arbitrary order of any Green's function of the theory. In practice, however, simplifications in computing the coefficients in the action and for operators can be made if only on-shell improvement is required. Here, only the Green's functions of physical states of the theory are improved. For on-shell improvement, the quantum creation operators need not be improved.

This Symanzik improvement scheme applied to the gauge action is examined in more detail in chapter 4.

1.2.2 Lattice Perturbation Theory

The usefulness of the lattice is associated with its ability to provide a regulation scheme and the accessibility of non-perturbative Monte-Carlo calculations. Perturbation theory is not redundant, however, as for asymptotically free theories, it provides useful information about the link to continuum physics and allows the dynamics of the fields at momenta higher than the lattice cut-off to be incorporated into calculations.

Like a continuum gauge theory, defined within a path integral formalism, lattice gauge theory has a natural perturbative expansion. In addition, the discretisation of spacetime changes the path integral into a finite integral and thus makes expansions finite. This implies the lattice has regulated the theory. A natural cut-off has been added, since modes with momenta higher than the inverse lattice spacing can not propagate. For a positive definite action, there exists a well defined vacuum for expansion.

Perturbation theory in the continuum provides reliable predictions of high energy

QCD processes. On the lattice, this implies that short-range lattice physics should be described accurately by perturbative expansions and thus the coefficients for Symanzik improvement of a local operator should be calculable in perturbation theory.

1.2.3 Tadpole Improvement

Using the bare lattice coupling, $\alpha_{\text{bare}} = \frac{g_{\text{bare}}^2}{4\pi}$ gives a perturbative expansion with poor convergence qualities. Typically, second order expansion terms are as large as the first-order terms for modest values of the coupling (≈ 0.08). The problem arises from the nature of the lattice link variables and the non-linear mapping to their continuum counterparts. In the continuum, the gauge fields are non-compact variables but the link variables, being members of the gauge group $SU(N)$ are necessarily compact. In a perturbative expansion, the two are related through

$$U_\mu(x) = \exp\left(iagA\mu(x)\right) \approx 1 + iagA_\mu(x). \tag{1.23}$$

But the higher terms, proportional to powers of agA_μ have UV divergences (from contributions due to tadpole graphs in the gluon propagator) whose effect is to exactly cancel the powers of the lattice spacing, a and so these terms only vanish as a power series in g as the continuum limit is approached.

In tadpole improving the links, the perturbative expansion is made about the mean-field value u_0

$$U_\mu(x) \approx u_0\left(1 + iagA_\mu(x)\right). \tag{1.24}$$

To preserve gauge invariance, u_0 is a constant multiplying the identity matrix. u_0 is determined non-perturbatively from a Monte-Carlo calculation of a UV quantity. In this thesis, the tadpole coefficient for $SU(3)$ is always defined in terms of the plaquette.

$$u_0 = \langle \frac{1}{3}\text{Re Tr}U_\square \rangle^{\frac{1}{4}}. \tag{1.25}$$

In practice then, the implementation of tree-level tadpole improvement is simple; in any operator on the lattice, the link variables are replaced with their tadpole improved counterparts.

$$\tilde{U}_\mu(x) = \frac{1}{u_0}U_\mu(x). \tag{1.26}$$

In practical computer simulations, it is convenient to keep the gauge link variables as members of the gauge group. The effects of tadpole improvement are then absorbed into rescalings of coupling constants in operators of interest. For simulations using the Wilson action, the replacement simply results in a renormalisation of the bare lattice coupling, $\beta \to \frac{\beta}{u_0^4}$, which does not affect simulations of the theory. For Symanzik-improved actions, built from a sum of traces over path-ordered products of different numbers of links, tadpole improvement leads to a relative shift in the perturbatively determined couplings in the action and is thus a non-trivial process.

1.3 Monte-Carlo Methods for Lattice Field Theories

Having discretised Euclidean spacetime onto a finite hypercubic lattice, the path integral has become a finite (but for realistic simulations, extremely large) dimensional integral. Expectation values of observables of the field theory on the lattice also have integral representations

$$\langle X \rangle_{\mathrm{true}} = \frac{1}{Z} \int \mathcal{D}U X[U] e^{-S[U]}, \tag{1.27}$$

with Z, the partition function of the theory given by

$$Z = \int \mathcal{D}U e^{-S[U]}. \tag{1.28}$$

The Monte-Carlo approach to solving these integrals is to generate a finite ensemble of N points in phase space (configurations) stochastically, with the probability of a point being selected, known and given by $P_{\mathrm{MC}}(U)$. With this ensemble, an approximation to the observable is then

$$\langle X \rangle_{\mathrm{MC}} = \frac{\sum_i^N X[U] P_{\mathrm{MC}}^{-1}[U] e^{-S[U]}}{\sum_i^N P_{\mathrm{MC}}^{-1}[U] e^{-S[U]}}. \tag{1.29}$$

This is known as importance sampling.

A natural simplification is made if the ensemble has the probability distribution generated by the action of the lattice theory.

$$P(U) = \frac{1}{Z} e^{-S[U]}. \tag{1.30}$$

Now the expectation value of an observable of the theory can be estimated by calculating the arithmetic mean over the set of configurations. The Monte-Carlo estimate on a set of N configurations, $\{U\}$ is then

$$\langle X\rangle_{\mathrm{MC}} = \frac{1}{N}\sum_{i}^{N} X[U]. \tag{1.31}$$

If the configurations are decorrelated, then the estimate obeys the central limit theorem and for sufficiently large N, approaches the true value of the theory with an error proportional to $\frac{1}{\sqrt{N}}$

$$\langle X\rangle_{\mathrm{MC}} = \langle X\rangle_{\mathrm{true}} + O(\frac{1}{\sqrt{N}}). \tag{1.32}$$

1.3.1 Markov Processes for Configuration Generation

An algorithm to generate an ensemble of configurations with the appropriate probabilistic weight of (1.30) is built with reference to the theory of Markov processes. A Markov process follows a random path through configuration space, generating a chain of configurations. The next member of the Markov chain is generated stochastically with reference to the current member only. Thus a transition rate for moving from configuration U to U', $R(U \to U')$ can be defined. After i updates, the probability of a particular configuration, U occuring in the ensemble is $P_M^{(i)}(U)$. A Markov chain can readily be constructed with fixed point probability equal to that of (1.30).

$$\lim_{i\to\infty} P_M^{(i)}(U) = P(U). \tag{1.33}$$

The transition rate is a normalised probability distribution and obeys

$$\int \mathcal{D}U' R(U \to U') = 1. \tag{1.34}$$

This is the simple statement that updating configuration U will generate some alternative configuration in phase space with probability 1.

To use a Markov process to generate the required ensemble, a number of rules must be applied in formulating the algorithm. A simplified discussion of these rules is presented here. For a more detailed exposition, see [1, 8].

An initial requirement is that the process is ergodic. A process is called ergodic if, on starting from any one point, there is a non-zero probability after a finite number of steps, of visiting any other point in configuration space. So there exists some finite L such that

$$R^{(L)}(U \rightarrow U') \neq 0. \tag{1.35}$$

Where $R^{(L)}$ is the transition probability for moving from configuration U to configuration U' after L repeated updates. Without ergodicity, the update algorithm breaks configuration space into disconnected subspaces and only samples configurations within the subspace containing the initial chain member.

It can be shown that a sufficient condition for (1.33) to hold is that the transition probability of an ergodic Markov process leaves a correctly distributed equilibrium ensemble invariant.

$$P(U') = \int \mathcal{D}U \; P(U)R(U \rightarrow U'). \tag{1.36}$$

This is the fixed-point equation of the chain.

A stronger constraint, which is sufficient to guarantee (1.36) holds, and which is imposed to help in the contruction of suitable update algorithms is that of detailed balance.

$$P(U)R(U \rightarrow U') = P(U')R(U' \rightarrow U). \tag{1.37}$$

The proof that processes obeying detailed balance also obey (1.36) follows directly from the definition and (1.34).

Note that a Markov process can be built from a set of different update algorithms providing they all possess a common fixed point. Here, the ergodicity requirement is relaxed to state that the compound set of updates must be capable of moving between two points in phase space. This is the case, for example, in the use of single-link update schemes. A set of N (N = the number of lattice degrees of freedom) different update processes are employed where the i^{th} process involves updating the i^{th} lattice degree of freedom while holding all the other variables fixed. Such a compound algorithm does not necessarily obey detailed balance, despite having the correct ensemble fixed point.

1.3.2 Heatbath Updating

A heatbath update scheme involves a direct refreshing of a stochastic variable from its equilibrium ensemble. This means the next member of the Markov chain is unrelated to its predecessor and is an independent sample for the Monte-Carlo integration. Unfortunately, for most lattice systems of interest, there is no method for simultaneously updating the entire set of lattice variables. However, a method for replacing a single variable can often be found and the lattice is updated by successively replacing each variable from its equilibrium ensemble whilst holding all the other degrees of freedom constant. This type of pseudo-heatbath update does leave correlations between successive chain members. A heatbath algorithm normally relies on mapping the probability distribution of interest onto the uniform probability in the range $(0, 1)$. Then, standard numerical techniques for generating random numbers of this type are employed and combined with the inverse map to generate the required probability. If a single variable, $x \in (a, b)$ has a normalised positive probability measure, $P(x)$ then

$$y(x) = \int_a^x P(z)\, dz \tag{1.38}$$

is a variable in the range $(0, 1)$ with a uniform distribution. If the integral of (1.38) can be solved and $x(y)$ calculated then a new value of the variable x can be generated from the uniform random variable, y.

Complex bosonic degrees of freedom with quadratic interactions can be updated from a heatbath. If the effective action for a single site is written as

$$\phi^*(x)M^\dagger M\phi(x) + \lambda^*(x)\phi(x) + \phi^*(x)\lambda(x), \tag{1.39}$$

then a new variable, η, given by

$$\eta = M\phi + [M^\dagger]^{-1}\lambda, \tag{1.40}$$

has the simple quadratic action $\eta^*\eta$ and can be generated from a gaussian ensemble using the Box-Müller algorithm. ($x_1, x_2 \in (0, 1)$ are uniformly distributed random numbers)

$$\eta = -\log(x_1)(\cos(2\pi x_2) + i\sin(2\pi x_2)). \tag{1.41}$$

Heatbath methods can be applied directly to single link updates of $SU(2)$ variables [9], when the effective probability for one link (with the other variables acting as a constant background) can be written as

$$P_{\text{eff}} = \frac{1}{Z_{\text{eff}}} e^{\beta Tr U_{\mu}(x) R_{\mu}(x)}. \tag{1.42}$$

For quenched gauge theories the force term, $R_{\mu}(x)$ can be calculated locally; it is the sum of staples around the link.

For $SU(3)$ (and in general $SU(N)$) a pseudo-heatbath method, due to Cabibbo and Marinari [10] can be applied to one link. Here, the variable is updated by three successive $SU(2)$ subgroup hits.

1.3.3 The Metropolis Test

One of the simplest forms of Markov updates is the Metropolis test [11]. In its most basic form, the algorithm is as follows; a new candidate configuration is proposed by making a random, reversible change to the existing configuration. Such a change obeys the condition

$$R(U \to U') = R(U' \to U). \tag{1.43}$$

The difference in the action of the two configurations, ΔS is then calculated, and the new candidate configuration is taken as the current member of the chain with probability

$$P_{\text{Metropolis}} = \min\left[1, e^{-\Delta S}\right]. \tag{1.44}$$

If the candidate is rejected, the current configuration becomes the new member of the chain. (1.43) and (1.44) together demonstrate that the Metropolis test obeys detailed balance for the action, S.

Using a small stochastic update step diffuses the configuration through phase space and so suffers from “random walk” correlations. The distance moved through phase space increases only as the square root of the number of updates. If a large update is applied (for example, globally updating all the links in a configuration) then for actions with large fluctuations the acceptance rate of the Metropolis test falls rapidly.

If an efficient algorithm to update configurations with a fixed point ensemble generated by

$$P_0[U] = \frac{1}{Z_0} e^{-S_0(U)}, \tag{1.45}$$

(where P_0 is a non-vanishing probability for all configurations) exists, then a modified version of the Metropolis algorithm can be employed to generate any ensemble from this update process. The proposed candidate configuration is generated reversibly from the S_0 update method and obeys detailed balance for the ensemble of (1.45).

$$R(U \to U')e^{-S_0(U)} = R(U' \to U)e^{-S_0(U')}. \tag{1.46}$$

Now the Metropolis test is on the relative change in the required fixed point action, S and the update action, S_0.

$$P_{\text{Metropolis}} = \min\left[1, e^{-(\Delta S - \Delta S_0)}\right]. \tag{1.47}$$

If the accessible action employed in performing the update step is approximately equivalent to the true action, then the fluctuations in $S - S_0$ are suppressed and the acceptance rate of the Metropolis will be higher than using the simple version. This principle is used in the construction of the exact algorithm of chapter 3.

1.3.4 Over-Relaxation

The Metropolis method and the heatbath method, when applied to a single lattice variable at a time, form techniques for executing a "random walk" through configuration space. For simulations with long correlation lengths (and the continuum limit for a field theory is reached in the limit of the correlation length of the system becoming infinite) this local random updating can lead to large autocorrelations between successive configurations. This is called "critical slowing down". An update scheme which deterministically moves the system through phase space will be more effective at updating long wavelength modes. Such an update scheme is called "Over-relaxation". Here, an action preserving, deterministic update is performed. Normally as with the heatbath method, this must be carried out for a single variable whilst holding all others fixed. This update scheme is not ergodic and must be used in conjunction with a stochastic scheme. Over-relaxation can be applied to complex boson fields with quadratic coupling and to $U(1)$ and $SU(2)$

gauge theories. $SU(3)$ over-relaxation is carried out as with the heatbath method by three $SU(2)$ subgroup hits.

1.3.5 The Markov Matrix and its Eigenvalues

For a system with a finite, discrete set of states, ψ (labelled by a single integer) the Markov transition probabilities form the elements of a matrix

$$R(i \to j) \equiv R_{ij}. \tag{1.48}$$

This allows the properties of a Markov chain of configurations of this system to be studied via matrix algebra. In particular the transition probability for L consecutive updates is found to be

$$R^{(L)}(i \to j) \equiv [R^L]_{ij}. \tag{1.49}$$

The steady state equation of (1.36) implies that if the equilibrium probability of state i is π_i then

$$R_{ij}\pi_j = \pi_i, \tag{1.50}$$

so that the equilibrium probability distribution is an eigenvector of the Markov matrix, with eigenvalue 1. All the other eigenvalues of the matrix must have modulus less than one. Hence any starting probability distribution will converge to the correct fixed point of the Markov chain.

The second and lower eigenvalues of the Markov matrix give information about the efficiency of the process in providing the Monte-Carlo estimator with decorrelated measurements.

1.3.6 Autocorrelations in Markov Chains

Since the next point in the chain is generated from its immediate neighbour, it is often the case that these two points will be close to each other in configuration space. This in turn implies that observables measured from neighbouring points on the Markov chain will be correlated and thus a sufficient number of updates must be performed on the configuration to generate a new sample point for the Monte-Carlo sum if the statistical limit of (1.32) is to hold. The efficiency of an algorithm is thus directly related to how effectively it moves through sample space with each update.

Consider again a system with a discrete set of states (as in section 1.3.5) labelled by integer values. The state of the system at time t is then ψ_t. The autocorrelation along the chain is defined as the probability that at two separated points on the chain, the system is in an identical configuration. So

$$A(\Delta t) = \sum_i \Pr(\psi_{t+\Delta t} = i | \psi_t = i) - \sum_i \Pr(\psi_{t+\Delta t} = i). \tag{1.51}$$

Using the matrix notation,

$$\Pr(\psi_{t+\Delta t} = i | \psi_t = i) = \left[R^{\Delta t}\right]_{ii} \text{ (no sum)} \tag{1.52}$$

and so (1.51) and (1.52) give

$$A(\Delta t) = \mathrm{Tr}\ R^{\Delta t} - 1. \tag{1.53}$$

If the Markov matrix is decomposed into its eigenvalues, $\{\lambda\}$, with the fixed point equation ensuring that the largest eigenvalue is 1, then

$$A(\Delta t) = \lambda_2^{\Delta t} + \lambda_3^{\Delta t} + \ldots\ . \tag{1.54}$$

For large separations on the chain, the autocorrelation falls exponentially with separation.

$$A(\Delta t) = \lambda_2^{\Delta t} = e^{\Delta t \ln \lambda_2}. \tag{1.55}$$

For a true heatbath algorithm, the eigenvalues of the Markov matrix are $\{1, 0, 0, \ldots\}$ and hence there are no autocorrelations between configurations. This will not necessarily be true for pseudo-heatbath techniques.

For Markov processes involving continuous variables, the Markov matrix has infinite dimension. The trace analysis above still applies but the eigenvalues of the matrix cannot be calculated analytically. In practice, autocorrelations in measurements in the theory are calculated by Monte-Carlo methods.

A measure of the efficiency of a practical update algorithm for lattice gauge theories is the autocorrelation time defined for each observable of the theory. The autocorrelation function for observable X calculated by a Markov process update algorithm is defined as

$$A_X(\Delta t) = \frac{\langle \tilde{X}[U_t]\tilde{X}[U_{t+\Delta t}]\rangle}{\langle \tilde{X}^2\rangle}, \tag{1.56}$$

with $\tilde{X}$, the deviation of X from its expectation value

$$\tilde{X}[U] = X[U] - \langle X\rangle. \tag{1.57}$$

The Markov analysis of the previous section predicts exponential decay in the autocorrelation function for large separations $((\frac{\lambda_2}{\lambda_3})^N \ll 1)$ along the chain

$$A_X(\Delta t) \approx e^{-\Delta t/\tau_{\mathrm{auto}}}, \tag{1.58}$$

with τ_{auto} the autocorrelation time for the observable of interest. For practical measurement of the autocorrelation behaviour of a Markov process, the integrated autocorrelation time is calculated.

$$\tau_{\mathrm{int}} = \sum_{\Delta t} A_X(\Delta t). \tag{1.59}$$

This process does not rely on measuring the large separation exponential decay and incorporates the contributions to autocorrelations from higher eigenvalues of the Markov matrix. Naturally, for a good Markov process, the autocorrelation time should be minimised, and so large separation measurements may require impractically large statistical measurements. If the autocorrelations in an observable are dominated by the second highest eigenvalues alone, then the autocorrelation time, defined by the exponential decay of A_X (1.58) and the integrated autocorrelation time (1.59) can be shown to be equivalent for large τ.

$$\tau_{\mathrm{int}} = \sum_{\Delta t} A_X(\Delta t) \approx \sum_{\Delta t} e^{-\Delta t/\tau_{\mathrm{auto}}} \text{ when } \tau_{\mathrm{auto}} \gg 1. \tag{1.60}$$

1.4 Computational Costs of Monte-Carlo Lattice

Calculations.

Monte-Carlo lattice calculations require large amounts of computer time. In this section, the principle source of these overheads is discussed with reference to $4d$ gauge theories such as QCD.

1.4.1 The Approach to the Continuum

So far, the foundation of a Monte-Carlo lattice calculation has been addressed; the generation of an ensemble of gauge field configurations. The computational cost of generating the configurations depends primarily on whether the quenched approximation has been employed. Quenched configurations can be generated with orders of magnitude less computer time than configurations including the effects of dynamical fermions. For fixed physical volume (a typical volume for a contemporary calculation in QCD is 2.5fm - for simulations with a realistic pion mass this would need to be larger) and correlation length, the cost of generating decorrelated configurations grows as

$$\text{Cost} \propto \left(\frac{1}{a}\right)^4 \left(\frac{1}{a^2}\right)^\alpha . \tag{1.61}$$

The first term arises simply from the increase in the number of lattice sites required as the lattice spacing is reduced at fixed volume; each site must be updated by the algorithm at some fixed computational cost per site. The second term arises as the lattice correlation length of the system grows as the continuum limit is approached. α is the *dynamical scaling exponent*, and depends on the details of the update process used. For algorithms such as the pseudo-heatbath method (Cabbibo-Marinari for $SU(N)$ gauge theories) where the system performs a random walk through phase space, this exponent is one. For updates encorporating over-relaxation methods, estimates of α range from between 0.5 and 1.

The next step in any calculations involving fermions (and this includes the majority of QCD processes of interest) is the generation of the fermion propagators. This contributes significantly to the cost of quenched calculations. The generation of propagators involves solving the linear equation

$$M\phi = \eta. \tag{1.62}$$

for some fermion source, η. This is normally carried out using a conjugate gradient or minimal residual solver [12]. Both these techniques are iterative; a better approximation to the true solution is made by some update and this update is performed repeatedly until some convergence criterion is met. Normally, convergence is based on the residue of the current best guess solution falling below some fixed value. The number of iterations required for convergence depends on both the lattice spacing and the pion mass (in a theory with a spontaneously broken chiral symmetry), since these govern the correlation length of the system in lattice units. Again, the cost of generating propagators grows at least as rapidly the number of lattice sites. In general,

$$\mathrm{Cost}_\phi \propto \left(\frac{1}{a}\right)^4 \left(\frac{1}{m_\pi^2 a}\right)^\gamma, \tag{1.63}$$

where γ depends on the solver used, but is around 1.

1.4.2 Dynamical Fermion Simulations

For dynamical fermion simulations, the fermion mass also influences the cost of the configuration generation step. To assess the cost of dynamical fermion simulations, the method for simulating the fermion determinant must be considered and the most commonly used algorithm, Hybrid Monte-Carlo [13], based on pseudofermion [14, 15] and molecular dynamics [16] techniques, is presented.

The fermion path-integral on the lattice is

$$\int \mathcal{D}\psi \mathcal{D}\bar{\psi}\; e^{-\bar{\psi}A\psi} = \det A. \tag{1.64}$$

The corresponding result for bosons, coupled through a positive definite matrix, B is

$$\int \mathcal{D}\phi \mathcal{D}\phi^*\; e^{-\phi^* B\phi} = \frac{1}{\det B}. \tag{1.65}$$

The boson path integral can be used to directly simulate an even number of fermion fields stochastically. A new set of bosonic degrees of freedom, called pseudofermions, are introduced. The pseudofermions are coupled non-locally through a positive definite matrix (the coupling matrix must be positive definite for (1.65) to hold)

$$\det M^2 = \det M^\dagger M = \int \mathcal{D}\phi\mathcal{D}\phi^* \; e^{-\phi^*[M^\dagger M]^{-1}\phi}. \qquad (1.66)$$

The entire lattice of pseudofermion variables can be readily updated from an exact heatbath. If a source of gaussian noise, η is generated, then ϕ can be refreshed as

$$\phi = M\eta. \qquad (1.67)$$

For staggered fermions, the matrix $M^2 \equiv -\not{D}^2 + m^2$ couples lattice sites of the same parity only. There are no direct even-odd or odd-even site couplings. The square root of the matrix of (1.66) can be taken by simulating pseudofermions on one lattice parity only [17]

$$\sqrt{\det M^2} = \det(\not{D} + m) = \int \mathcal{D}\phi_p\mathcal{D}\phi_p^* e^{-\phi_p^*[\not{D}^2+m^2]_{pp}^{-1}\phi_p}. \qquad (1.68)$$

The heatbath result of (1.67) can be extended to incorporate the reduced number of flavours. To refresh the pseudofermions on one parity from a heatbath, a noise field on both parities $(\eta_p, \eta_{\bar{p}})$ is generated, then

$$\phi_p = \not{D}_{p\bar{p}}\eta_{\bar{p}} + m\eta_p. \qquad (1.69)$$

The full partition function for $2^{\frac{d}{2}}$ flavours of fermions interacting with the gauge fields is now

$$Z = \int \mathcal{D}U\mathcal{D}\phi_p\mathcal{D}\phi_p^* e^{-S_G(U)-\phi_p^*[\not{D}^2+m^2]_{pp}^{-1}\phi_p}. \qquad (1.70)$$

The pseudofermion action induces non-local gauge field interactions. If a single link on the lattice is updated, a global lattice calculation of the new pseudofermion action would need to be performed, implying that the cost of such an update method would grow at least as $(\frac{1}{a})^8$. For efficient Markov processes to exploit the pseudofermion technique, they must carry out global updates of all the lattice sites between recalculation of the pseudofermion action. Such a set of global update schemes exist. The molecular-dynamics and Hybrid Monte-Carlo algorithms are examples of global update schemes.

In the molecular dynamics scheme, the gauge link variables are considered as describing the position of particles with an interaction potential given by the action of (1.70). A fictitous time variable, τ, is introduced along with canonical momenta conjugate to the link variables. These are defined such that

$$\frac{dU_\alpha(\tau)}{d\tau} = ip_\alpha(\tau), \tag{1.71}$$

with the index α denoting the spacetime coordinate and link direction. The momenta are generated from an appropriate gaussian ensemble. The molecular dynamics system then introduces a classical Hamiltonian describing evolution in simulation time, τ,

$$H = \frac{1}{2}p_\alpha^2 + S(U, \phi, \phi^*), \tag{1.72}$$

and the equations of motion can be derived by ensuring conservation of the Hamiltonian with the system's progress through simulation time. An expectation value of some observable of the gauge theory is then given via the ergodic hypothesis as the time average of the observable measured on the system as it evolves. To ensure ergodicity, from time-to-time the conjugate momenta are refreshed from a heatbath. In practice, the integration of the equations of motion must be performed numerically by finite stepsize methods, for example, using the leapfrog algorithm. This introduces systematic errors into measurements; for first-order leapfrog, these errors are $O(\delta\tau^2)$. To reliably calculate expectation values, simulations must be performed at a range of stepsizes and results extrapolated back to zero stepsize.

This extrapolation can be avoided by using an extension to the molecular dynamics scheme; Hybrid Monte-Carlo. The finite stepsize errors can be removed by adding to the Markov process an accept/reject step on the change in the Hamiltonian. If the equations of motion are perfectly integrated, the Hamiltonian of (1.72) will be conserved and the Metropolis test will always accept.

A summary of the HMC algorithm is

1. Start with an inital configuration, $\{U\}$.

2. Refresh the pseudofermions (in equilibrium with the current gauge configuration) from a heatbath.

3. Refresh the conjugate momenta from a heatbath.

4. Store the present gauge configuration, $\{U\}$.

5. Calculate the Hamiltonian of (1.72)

6. Integrate the equations of motion using a reversible (leapfrog) scheme, for n steps of length $\delta\tau$.

7. Recalculate the Hamiltonian.

8. Perform a Metropolis accept/reject test on the change in the Hamiltonian, $P_{\text{acc}} = \min\left[1, e^{-\delta H}\right]$. If the change is rejected, restore the configuration to the one stored in step 4, else proceed with the new configuration.

9. Return to step 2.

The Hybrid Monte-Carlo algorithm is exact for any choice of stepsize. For this reason, it is the most popular algorithm in use in present unquenched simulations involving an even number of fermion flavours.

To estimate the cost of approaching the continuum of full QCD is a difficult art. For HMC [18], there are many significant factors to consider. The dominant overhead comes from the need to invert $M^\dagger M$ for every leapfrog integration step. The number of iterations for adequate solver convergence rises roughly as the inverse fermion mass. The acceptance rate of the global Metropolis step is dependent on fluctuations in the Hamiltonian due to finite stepsize integration errors. To maintain a good acceptance rate, the stepsize for the molecular dynamics process must be of the same order as the period of the highest frequency mode of the system. This implies that the stepsize should be proportional to the fermion mass. As a rule of thumb, the length of molecular dynamics time the system is evolved through between accept/reject decisions should be about one, and hence the number of leapfrog steps increases as the inverse of the fermion mass. An estimate for the cost of HMC updates [19, 20] gives

$$\text{Cost}_{\text{HMC}} \propto \left(\frac{1}{a}\right)^5 m_f^{-13/4}. \tag{1.73}$$

1.5 Summary

In this chapter, some aspects of formulating gauge theories on the lattice have been addressed. Attention has focussed on the lowest level process in a Monte-Carlo calculation; configuration generation. Along with the evaluation of fermion propagators, with which it is inextricably linked, this process accounts for the dominant computational cost of any lattice calculation. Two of the major sources of the CPU overheads have been discussed; the approach to the continuum limit ($a \to 0$) and the inclusion of dynamical fermions in the configuration generation process. For both quenched and dynamical simulations, the approach to the continuum limit has been shown to be very costly (at least $1/a^5$) and thus reliable calculations at large lattice spacings seem desirable. Dynamical simulations add extra complexity at the basic algorithm level and suffer from worse critical slowing down, and new methods for these simulations should be considered.

The remaining chapters of this thesis discuss potential improvements to configuration generation methods. Chapters 2 and 3 address the problem of dynamical fermion simulations by testing and extending a new bosonisation system. Chapter 4 discusses improvements to the discretisation scheme of lattice QCD, with an aim to performing reliable calculations on a coarse lattice, thus avoiding the difficult approach to the continuum.

Chapter 2

Dynamical Fermion Simulations Using the Local Bosonic Action

In this chapter, the behaviour of an algorithm proposed by Lüscher [21] for simulating the unquenched partition function of a lattice gauge theory with an even number of flavours of fermions is described. The method exploits an approximate mapping which converts the determinant of the fermion matrix into a partition function of a large number of flavours of auxiliary bosons. The mapping becomes exact in the limit of an infinite number of auxiliary boson fields. The advantage is that the resulting theory has only local, bosonic interactions, making it amenable to a wide range of Monte-Carlo simulation techniques. Results from a study of the reliability of the approximation employed and a test of the method's performance as a Monte-Carlo configuration generation algorithm are presented.

The problem of Monte-Carlo calculations of lattice gauge theories with dynamical fermions is inextricably linked to that of inverting the coupling matrix present in the action. In the HMC algorithm, this link involves the use of pseudofermions to mimic the fermion determinant, requiring an inversion of the interaction matrix with every molecular dynamics step performed. For the method examined in this chapter, contact with inversion techniques is made at an early stage.

2.1 The Local Bosonic Action Algorithm for Dynamical Fermions

Before describing the algorithm for dynamical fermion simulations, polynomial approximations to inverses are introduced. These approximations form the basis for the method.

2.1.1 Polynomial Approximate Matrix Inversion

Consider a polynomial approximation of even order, n, $\mathcal{P}(s)$ to $1/s$, which is valid in some bounded range

$$\mathcal{P}(s) = \sum_{i=0}^{n} c_i s^i. \tag{2.1}$$

For simplicity, the range of validity is assumed to be the interval $0 < s \leq 1$. Such an n^{th} order polynomial can be uniquely classified in terms of its n complex roots, $\{z_k\}$. The polynomial can then be written

$$\mathcal{P}(s) = c_n \prod_{k=1}^{n} (s - z_k). \tag{2.2}$$

If the number of polynomial terms employed is to be kept to a minimum (and this is crucial for the fermion algorithm) then rapid convergence of the polynomial approximation is required. To this end, a Chebyshev acceleration scheme is employed. Defining the error in the polynomial as

$$\mathcal{R}(s) = 1 - s\mathcal{P}(s), \tag{2.3}$$

then since $\mathcal{P}(s)$ is a polynomial in positive powers of s, it follows $\mathcal{R}(0) = 1$. A parameter, ϵ which sets the lower scale of the approximation is then introduced. Now the $(n+1)^{\text{th}}$ order error polynomial, $\mathcal{R}(s)$ is chosen to have a minimum upper bound, δ in the range $\epsilon \leq s \leq 1$.

$$\delta = \max_{\epsilon \leq s \leq 1} |\mathcal{R}(s)|. \tag{2.4}$$

To minimise δ, it can be shown [21, 22] that $\mathcal{R}(s)$ must be a scaled, translated Chebyshev polynomial

$$\mathcal{R}(s) = \frac{T_{n+1}(\frac{2s}{1-\epsilon} - \frac{1+\epsilon}{1-\epsilon})}{T_{n+1}(-\frac{1+\epsilon}{1-\epsilon})}. \tag{2.5}$$

The error bound, δ is then

$$\delta = \frac{1}{T_{n+1}(-\frac{1+\epsilon}{1-\epsilon})} \approx 2\left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)^{n+1} \approx 2e^{-2n\sqrt{\epsilon}}. \tag{2.6}$$

The zeros of the polynomial, $\mathcal{P}$ are given by noting $T_n(s) = \cos(n\cos^{-1}(s))$ and lie on an ellipse (see figure 2.1)

$$z_k = \frac{1+\epsilon}{2}\left\{1 - \cos\left(\frac{2\pi k}{n+1}\right)\right\} - i\sqrt{\epsilon}\sin\left(\frac{2\pi k}{n+1}\right) \quad (k = 1\dots n).. \tag{2.7}$$

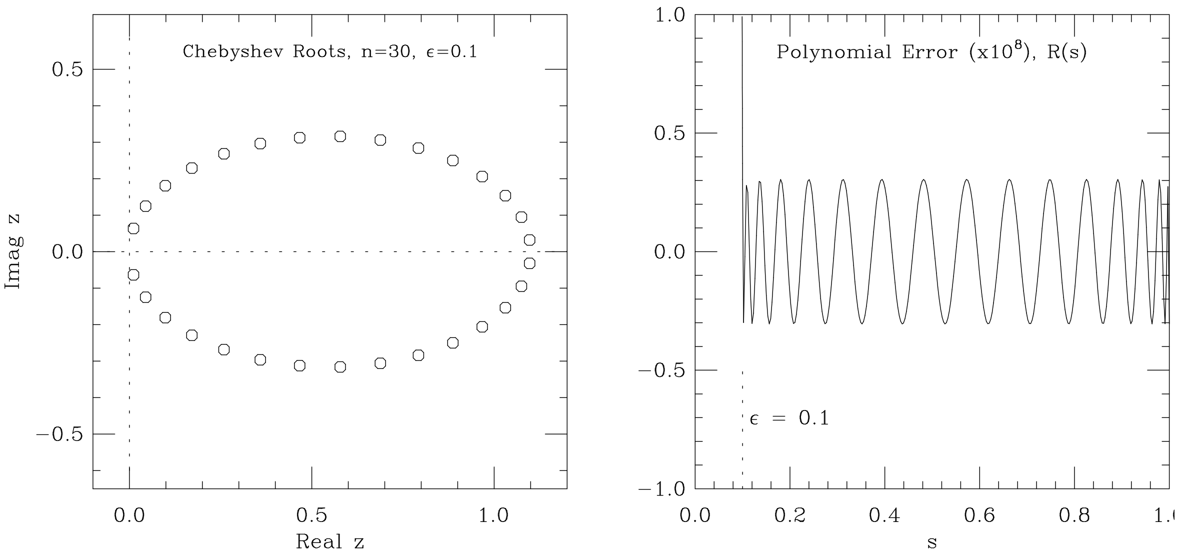


Figure 2.1: The Chebyshev polynomial. Left - The roots of the polynomial for $n = 30, \epsilon = 0.1$. Right - The error function, $\mathcal{R}(s)$ (magnified by 10^8) for these roots

Polynomial approximations to the inverse of hermitian matrices with bounded eigenvalues can be made by a direct analogy. If an hermitian matrix, H has eigenvalues in the range $0 < \lambda \leq 1$ then an approximation to the inverse is

$$H^{-1} \approx \mathcal{P}(H) = c_n \prod_{k=1}^{n}(H - z_k). \tag{2.8}$$

The nature of the inverse approximation of this matrix polynomial can be seen by a diagonalisation of H.

For roots generated with Chebyshev acceleration techniques, it is important if the convergence is to be optimal to select the lower bound parameter, ϵ so the smallest eigenvalue of the matrix lies in the range $\epsilon < \lambda_{\min} < 1$. Equation (2.6) suggests ϵ should be as large as possible to reduce the error bound and thus the best choice for the parameter is $\epsilon \approx \lambda_{\min}$. This tuning of the convergence parameter will be discussed in the context of fermion matrix inversion later in this chapter.

2.1.2 Fermion Matrix Inversion

The techniques of Chebyshev accelerated polynomial inversion can be directly applied to the inversion of the fermion matrix of lattice gauge theories.

The matrix for staggered fermions of finite mass has no definite hermiticity. However, a transformation that preserves the determinant in the path integral, whilst making the matrix hermitian is;

$$Q = \Gamma_5(\not{D} + m) \tag{2.9}$$

where Γ_5 is the lattice staggered version of γ_5, the Dirac matrix,

$$\Gamma_5 = \begin{pmatrix} 1_{ee} & 0 \\ 0 & -1_{oo} \end{pmatrix}. \tag{2.10}$$

This implies $\det Q = \det \Gamma_5 \det(\not{D}+m) = \det(\not{D}+m)$. The matrix Q^2 is hermitian and positive definite with eigenvalues in the range $m^2 \leq \lambda \leq d^2 + m^2$. A rescaled hermitian matrix,

$$\tilde{Q}^2 = \frac{Q^2}{(d^2 + m^2)}, \tag{2.11}$$

has eigenvalues in the range $0 \leq \lambda \leq 1$ for any value of the bare fermion mass. With a matrix in this form, the polynomial analysis of section 2.1.1 can be applied to staggered fermion matrix inversion.

2.1.3 The Local Bosonic Action for Dynamical Fermions

The partition function of a lattice gauge theory with an even number of flavours of dynamical fermions (after analytic integration of the fermionic degrees of freedom) is

$$Z = \int \mathcal{D}U \det(\not{D} + m)^2 e^{-S_G[U]} \tag{2.12}$$

The number of fermion flavours is $2^{\frac{d}{2}+1}$, d =number of spacetime dimensions.

The fermion matrix in the determinant can be replaced by the rescaled hermitian form of (2.11). This simply rescales the partition function by an overall constant. $\tilde{Q}^2$ has the properties required for the analysis of section 2.1.1. A polynomial approximation to the inverse of $\tilde{Q}^2$ is

$$\tilde{Q}^{-2} \approx \mathcal{P}(\tilde{Q}^2) = c_n \prod_{k=1}^{n} (\tilde{Q}^2 - z_k), \tag{2.13}$$

and so

$$\det \tilde{Q}^2 \approx \frac{1}{\det \mathcal{P}(\tilde{Q}^2)} = \frac{1}{c_n^V \det \prod_{k=1}^{n} (\tilde{Q}^2 - z_k)}. \tag{2.14}$$

The roots of the polynomial come in complex conjugate pairs, and so the product of matrices in the denominator can be rewritten

$$\begin{aligned} \prod_{k=1}^{n} (\tilde{Q}^2 - z_k) &= \prod_{k=1}^{n/2} (\tilde{Q}^2 - z_k)(\tilde{Q}^2 - z_k^*) & (2.15) \\ &= \prod_{k=1}^{n/2} (\tilde{Q} - \sqrt{z_k})(\tilde{Q} + \sqrt{z_k})(\tilde{Q} - \sqrt{z_k^*})(\tilde{Q} + \sqrt{z_k^*}). & (2.16) \end{aligned}$$

The hermitian matrix, Q has eigenvalues in $\pm$ pairs so it can be shown that, for any complex w,

$$\det(\tilde{Q} - w) = \det(\tilde{Q} + w). \tag{2.17}$$

Exploiting this degeneracy, the determinant of the product becomes

$$\det \prod_{k=1}^{n} (\tilde{Q}^2 - z_k) = (\prod_{k=1}^{n/2} \det\{(\tilde{Q} - \sqrt{z_k})(\tilde{Q} - \sqrt{z_k^*})\})^2. \tag{2.18}$$

The number of fermion flavours in the simulation can be halved to $2^{\frac{d}{2}}$ by using the degeneracy to take the square root of the determinant of (2.12). The partition function for $2^{\frac{d}{2}}$ is approximately simulated by

$$Z_{boson} = \int \mathcal{D}U \frac{1}{\prod_{k=1}^{n/2} \det\{(\tilde{Q} - \sqrt{z_k})(\tilde{Q} - \sqrt{z_k^*})\}} e^{-S_G[U]}. \tag{2.19}$$

The transformation to a theory with $n/2$ flavours of bosons is made by noting that each term in the product of determinants in (2.19) is a positive definite matrix and can thus be replaced by the path integral of a field of bosons (see (1.65)). For one term,

$$\frac{1}{\det\{(\tilde{Q} - \sqrt{z_k^*})(\tilde{Q} - \sqrt{z_k})\}} = \int \mathcal{D}\phi_k \mathcal{D}\phi_k^* \, e^{-\phi_k^*(\tilde{Q}-\sqrt{z_k^*})(\tilde{Q}-\sqrt{z_k})\phi_k} \tag{2.20}$$

and then the approximate partition function is

$$Z_{boson} = \int \mathcal{D}U \mathcal{D}\phi_k \mathcal{D}\phi_k^* \, e^{-S_G[U]-S_\phi[U,\phi_k,\phi_k^*]}, \tag{2.21}$$

where the boson action is

$$S_\phi[U,\phi_k,\phi_k^*] = \sum_k \phi_k^*(Q-y_k^*)(Q-y_k)\phi_k. \tag{2.22}$$

Here, a rescaled variable, $y_k = \sqrt{(d^2+m^2)z_k}$ is defined so the unscaled, hermitian fermion matrix, Q explicitly appears in the action.

This is the action of a theory, invariant under gauge-transformation, of interacting gauge bosons and n, the degree of the polynomial employed (or $n/2$ for half the fermion flavours) auxiliary boson fields. The non-local fermion determinant has been substituted for a local bosonic action, which makes Monte-Carlo simulation amenable to a wider range of existing algorithms. The partition function is, however, an approximation to the full fermionic theory and the validity of this step is discussed in section 2.5.

2.2 The Schwinger Model

To test the performance of the algorithm and the accuracy of the approximation, the lattice Schwinger model [23] is used. The Schwinger model is $2d$ QED (which has a $U(1)$ gauge group).

The lattice version of the theory is confining and asymptotically free, so results relating to the performance of the algorithm should be similar to simulations of more physically interesting lattice theories such as QCD. The local boson algorithm requires an even number of flavours of fermions and the minimum number that can be simulated is $2^{\frac{d}{2}}$. For the Schwinger model, the method simulates two flavours.

The gauge link variables, $U_\mu(x), \mu = 1,2$ are members of the compact, abelian group $U(1)$. They can be related directly to the bosonic field representations by exponentiation

$$U_\mu(x) = e^{-iagA_\mu(x)} = e^{-i\theta_\mu(x)}. \tag{2.23}$$

The gauge field kinetic term in the action is

$$S_G[U] = -\beta \sum_{\square} \cos \theta_{\square}(x), \tag{2.24}$$

with

$$\theta_{\square}(x) = \theta_1(x) + \theta_2(x+\hat{1}) - \theta_1(x+\hat{2}) - \theta_2(x). \tag{2.25}$$

The staggered matrix for two flavours of fermions of bare mass, m coupled to the $U(1)$ gauge group is

$$M_{x,y}[U] = m\delta_{x,y} + \frac{1}{2}\sum_{\mu} \eta_\mu(x) \left(\delta_{x+\mu,y} e^{-i\theta_\mu(x)} - \delta_{x-\mu,y} e^{i\theta_\mu(x-\mu)}\right) \tag{2.26}$$

where the Dirac structure is contained in the staggered phases, $\eta_\mu(x)$

$$\eta_1(x) = 1, \quad \eta_2(x) = -1^{x_1}. \tag{2.27}$$

2.2.1 Topological Charge and Approximate Zero Modes

A lattice definition of a topological charge for the Schwinger model with periodic boundary conditions can be made [24]. First, defining the lattice field strength, F(x), by

$$e^{iF(x)} = e^{i\theta_{\square}(x)} \text{ with } F(x) \text{ restricted to } -\pi \le F(x) < \pi \tag{2.28}$$

the topological charge on a given configuration is then

$$Q_{\text{top}} = \frac{1}{2\pi}\sum_x F(x) \tag{2.29}$$

and it can be shown from this definition that $Q_{\text{top}} \in \mathbf{Z}$. Associated with the topologically charged sectors of the gauge field are approximate zero modes (AZM) of the Dirac operator [25]. Here, the operator has $|Q_{\text{top}}|$ AZM's on a charged sector. This result is the lattice remnant of a continuum (Atiyah-Singer) index theorem [26] for the model on a torus.

This behaviour makes the model a useful platform for testing Lüscher's algorithm for two reasons. Firstly, the topologically charged modes are suppressed by the presence of dynamical fermions. The AZM on a charged configuration means the fermion determinant is smaller here, making the probabilistic weight for the configuration small. With Lüscher's method, the breakdown of the polynomial below the scale ϵ means the bosonic determinant of (2.19) underestimates the suppression of the charged sectors. Any observable strongly linked to the presence of these charged modes (such as the smallest eigenvalue of the fermion matrix) could have a large error induced by this effect. Secondly, the charged sectors are separated by a high potential barrier which means Monte-Carlo algorithms often have long tunnelling times between these sectors leading to high autocorrelation times for observables linked with the charge [27]. This charge tunneling is used in making a direct comparison between the modified Lüscher algorithm of chapter 3 and HMC.

2.3 Free Field Analysis

The eigenvalues of the fermion matrix Q^2 can be studied analytically for the free fermion case. Some useful information relating to the behaviour of Chebyshev polynomial approximations can be gained by study of their application to this case.

An error function for approximations to the inverse of the free fermion matrix of mass, m_f on a lattice of extent L is defined as

$$E(m_f, L; n, \epsilon) = \frac{1}{L^2} \mathrm{Tr} \left(1 - \tilde{Q}^2 \mathcal{P}(\tilde{Q}^2)\right)^2 . \tag{2.30}$$

For sufficiently heavy fermions (so finite volume effects are small), this has weak L dependence.

The error for fermions of mass, $m = 0.1$ as a function of ϵ for $n = 10, 20, 30, 40$ and 50 is shown in figure 2.2. The value of the Chebyshev parameter ϵ which minimises the error, ϵ_{opt} is computed for all values of n between 10 and 50. These values are indicated on the figure by the curved line and can be seen to be greater than the smallest eigenvalue of the scaled fermion matrix, $\tilde{Q}^2$ for a finite number of polynomial terms, and approaches this value asymptotically in the exact limit ($n \to \infty$). This implies that, for dynamical simulations using the bosonic action, the Chebyshev parameter should be set slightly above the expected smallest eigenvalue of the fermion matrix.

Figure 2.3 demonstrates the effects on the error function of running the number of polynomial terms and free fermion mass. The optimised value of ϵ is employed in each instant. The two graphs demonstrate the expected behaviour for Chebyshev accelerated polynomial approximations; an exponential fall in the error with the number of terms. Also note that from the second graph, there is an exponential fall in the error as the fermion mass increases, so

$$\log E \propto -nm_f a. \tag{2.31}$$

In order to preserve a constant error, the number of polynomial terms needed should rise as the inverse of the fermion mass

$$n \propto \frac{1}{m_f}. \tag{2.32}$$

This prediction is used later to determine the critical scaling of the algorithm with the fermion mass. For a theory in a spontaneously broken chirally symmetric phase (such as QCD) the Goldstone mode is the pseudoscalar state (in QCD, the pion) and the prediction from chiral symmetry is

$$m_P^2 \propto m_f. \tag{2.33}$$

Hence, expressed in terms of the (more relevant physically for simulations) pseudoscalar mass

$$\log E \propto -nm_P^2 a. \tag{2.34}$$

and so the number of boson field (or terms in the polynomial) must increase with the inverse pseudoscalar mass squared.

From (2.6) the polynomial error bound estimate predicts that, in order to preserve a constant error, the Chebyshev parameter scales with number of polynomials as

$$\epsilon \propto \frac{1}{n^2}, \tag{2.35}$$

and this estimate is borne out by examining the optimised value of Chebyshev parameter in the free field analysis. This result is also used later.

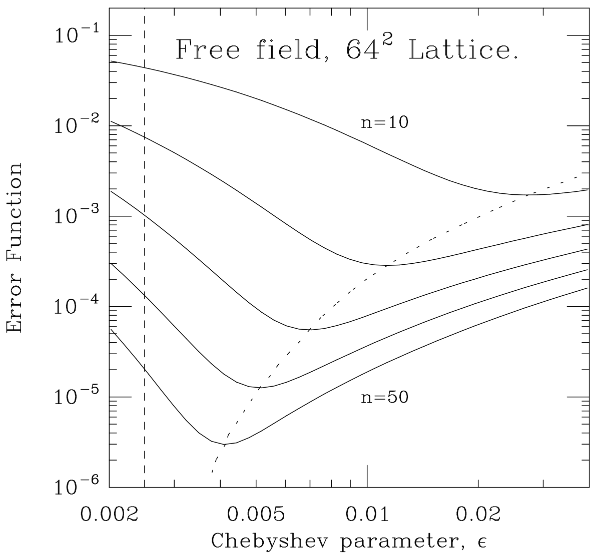


Figure 2.2: The ϵ dependence of the error function for free fermions, m=0.1. The lines are for $n = 10 \dots 50$ polynomial terms. The vertical line indicates the value of $\lambda_{\mathbf{min}}[\tilde{Q}^2]$ at this mass. The dashed line indicates the position of the minimum of the error function at each polynomial order.

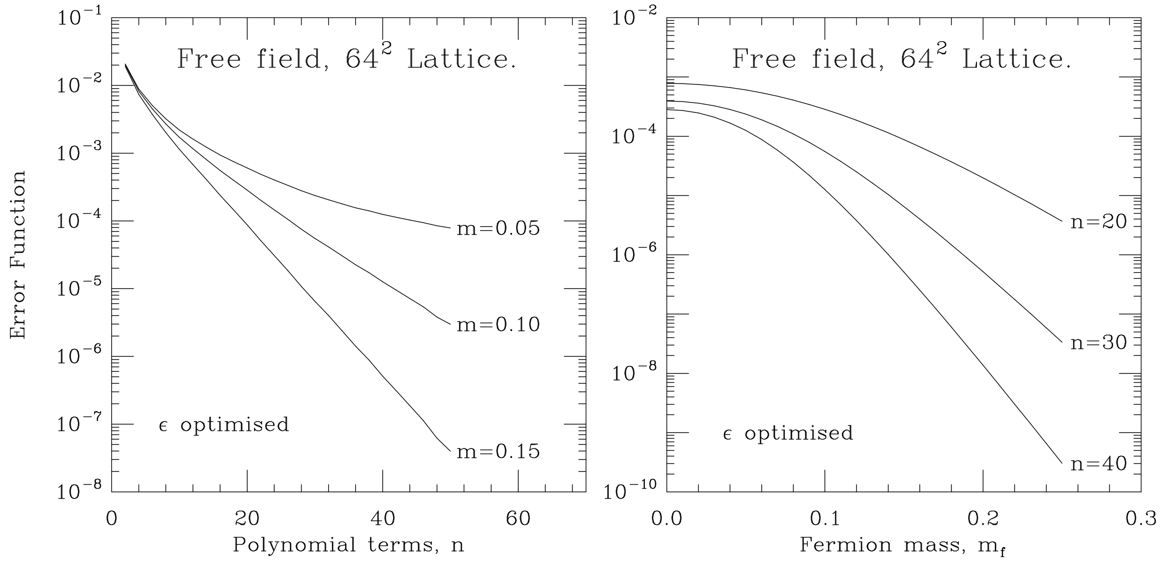


Figure 2.3: The error function dependence on the number of polynomial terms and free fermion mass. In both cases, the results use the optimised value of ϵ.

2.4 Monte-Carlo Implementation of the Algorithm

The action of 2.22 is local, bounded below, and positive definite and can thus be simulated by Monte-Carlo methods.

2.4.1 The Auxiliary Boson Fields

There is no direct coupling between different flavours of auxiliary boson fields (but they do couple indirectly via the gauge fields) so the change in the action resulting from changes made to one field at one site is a local computation, and references only one flavour of the auxiliary fields. Boson field updates then require CPU time that grows linearly in the number of boson fields. This makes stochastic simulation of the bosonic partition function straightforward. The simplest implementation is to update the boson fields site-by-site from a local gaussian heatbath.

For changes to boson fields at one site, the quadratic action is

$$S_{\phi\mathrm{eff}}(x) = -c_k(x)\phi_k^*(x)\phi_k(x) + \lambda^*(x)\phi_k(x) + \phi_k^*(x)\lambda(x). \tag{2.36}$$

The "background", λ depends on the neighbouring boson sites (see figure 2.4) and the local configuration of gauge links. c_k is independent of the configuration but has site dependence since the staggered fermion matrix splits spin components over sites.

$$c_k = \begin{cases} |m - y_k|^2 + 1 & \text{On even sites} \\ |m + y_k|^2 + 1 & \text{On odd sites} \end{cases} \tag{2.37}$$

With this action, the field on the site can be updated from a heatbath. If η is stochastic gaussian noise, then the update is

$$\phi_k(x) \to \frac{1}{\sqrt{c_k(x)}}\left(\eta - \frac{1}{\sqrt{c_k(x)}}\lambda(x)\right). \tag{2.38}$$

2.4.2 The Gauge Fields

For the gauge fields, the change in the action for an update to one link can be calculated locally (see figure 2.4) and has two contributions; from the gauge field kinetic term and from the local boson action

$$S_{G\text{eff}}(x) = S_{PG}(x) + S_{\text{Boson}}(x). \tag{2.39}$$

This can be reparameterised in terms of two variables, $\beta_{\text{eff}}(x)$ and $\Delta(x)$ to the simple form

$$S_{G\text{eff}}(x) = -\beta_{\text{eff}}(x)\left(\cos(\theta(x) - \Delta(x)\right). \tag{2.40}$$

For efficiency, an effective force term relating the change in the auxiliary bosonic action to link updates can be computed and the influence of all $n/2$ flavours of bosons summed externally to these updates. This force term need only be updated every time the auxiliary bosons are altered allowing for multiple hits to the gauge fields, without reference to the boson configurations which are then (in terms of CPU time) independent of the number of boson fields.

For light bare fermion masses, where fluctuations in the boson action are large compared with the gauge action, the most efficient update for gauge links is an over-relaxation technique. Here, an update is proposed which leaves the action unchanged. From 2.40 the over-relaxation update is

$$\theta(x) \rightarrow 2\Delta(x) - \theta(x). \tag{2.41}$$

Ergodicity of the method is ensured by stochastic updates of the coupled auxiliary boson fields. The over-relaxation technique has the advantage of computational simplicity and good autocorrelation performance; since it is a deterministic step it does not suffer from "random walk" autocorrelations (see section (1.3.4)).

This pure over-relaxation method becomes less effective for heavier fermion masses as the fluctuations in the fermion determinant (and hence the auxiliary boson action) are reduced. Since the only changes in the pure gauge action are via exchange of energy with the auxiliary bosons, this route is suppressed. For heavier fermion masses, an additional stochastic update for the gauge fields must be employed. The simplest is a Metropolis update.

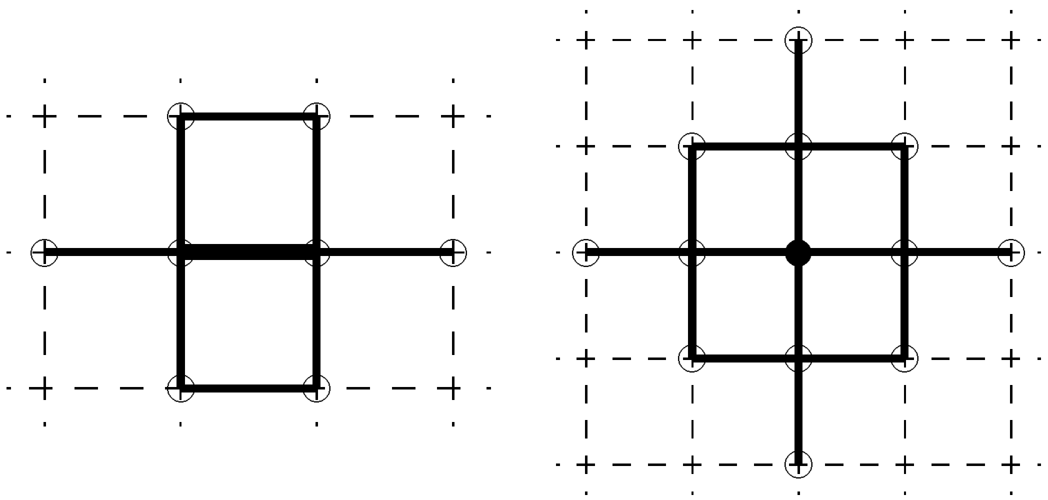


Figure 2.4: The Lüscher Local Interactions; Left - The central gauge link (bold) is being updated. Right - The central boson field (full circle) is being updated. Direct interactions are with the gauge fields (lines) and Lüscher boson fields (circles).

2.5 Results - The Approximation

Dynamical simulations of the Schwinger model with light fermions were performed to check the validity of the local bosonic action approximation to the full theory. The results were directly compared with a HMC simulation. The local boson action was tested with a range of numbers of auxiliary boson flavours ranging from 10 to 25 (corresponding to polynomials of degree 20 to 50) and the approach to the true value of the theory (calculated from the HMC simulation) was monitored.

The simulations were performed at $\beta = 3.0$ with two flavours of fermions of mass, $m = 0.1$ (in lattice units). For the Schwinger model, this is sufficiently far from the strong coupling regime to give reliable predictions for the algorithm's performance in the approach to continuum limit of the theory. The simulations were run on 16×16 lattices. For all the Lüscher simulations, the Chebyshev acceleration parameter was fixed to $\epsilon = 0.003$. This choice was made with reference to the free field analysis of section 2.3, which suggests the parameter should be set slightly above the smallest eigenvalue of $\tilde{Q}^2$ (here about 0.0025).

Figure 2.5 shows the expectation value of the plaquette for approximate dynamical simulations using different numbers of polynomial terms. The horizontal, dashed lines are the true dynamical value, generated from simulations using the HMC algorithm. The important point to notice from figure 2.5 is the non-monotonic approach towards the "true" theoretical value. Note the expectation value of the plaquette in the quenched theory which corresponds to the local bosonic action with a polynomial approximation of degree $n = 0$ is 0.8100 (calculated analytically for infinite volume) and yet the result for the $n = 20$ simulation is above the HMC estimate. This effect has been noted in other theories and may pose a serious problem to extrapolation to the true theoretical result from simulations with accessible numbers of auxiliary bosons.

Figure 2.6 is the expectation value of the lowest eigenvalue of $\not{D}^2$. The approximation underestimates the eigenvalue. This result is to be anticipated since the error in the polynomial means the probabilistic weight for configurations with low $\not{D}$ eigenvalues is over-estimated.

The expectation value in the full theory can be calculated by including the effects of the error in the polynomial [28]. This leads to

$$\langle A \rangle_{\text{full}} = \frac{\langle A \ \det \tilde{Q}^2 \mathcal{P}(\tilde{Q}^2) \rangle_{\text{Luscher}}}{\langle \det \tilde{Q}^2 \mathcal{P}(\tilde{Q}^2) \rangle_{\text{Luscher}}}. \tag{2.42}$$

The approach to the correct value of the full theory is then dependent on the correlation between the operator of interest and the error determinant of (2.42) which differs for each operator. The correction can be done explicitly as a reweighting of the Monte-Carlo result, but this leads to an increase in statistical fluctuation and the determinant calculation would be prohibitively expensive for $4d$ theories, such as QCD.

The expectation value of an unquenched theory with light fermions has, however been calculated accurately (such that the systematic errors are less than statistical errors) with an accessible number of auxiliary boson fields. It is a necessary condition for simulating QCD that reliable results can be obtained with $n \approx 50$ if the algorithm is to be used to simulate QCD on modern supercomputers.

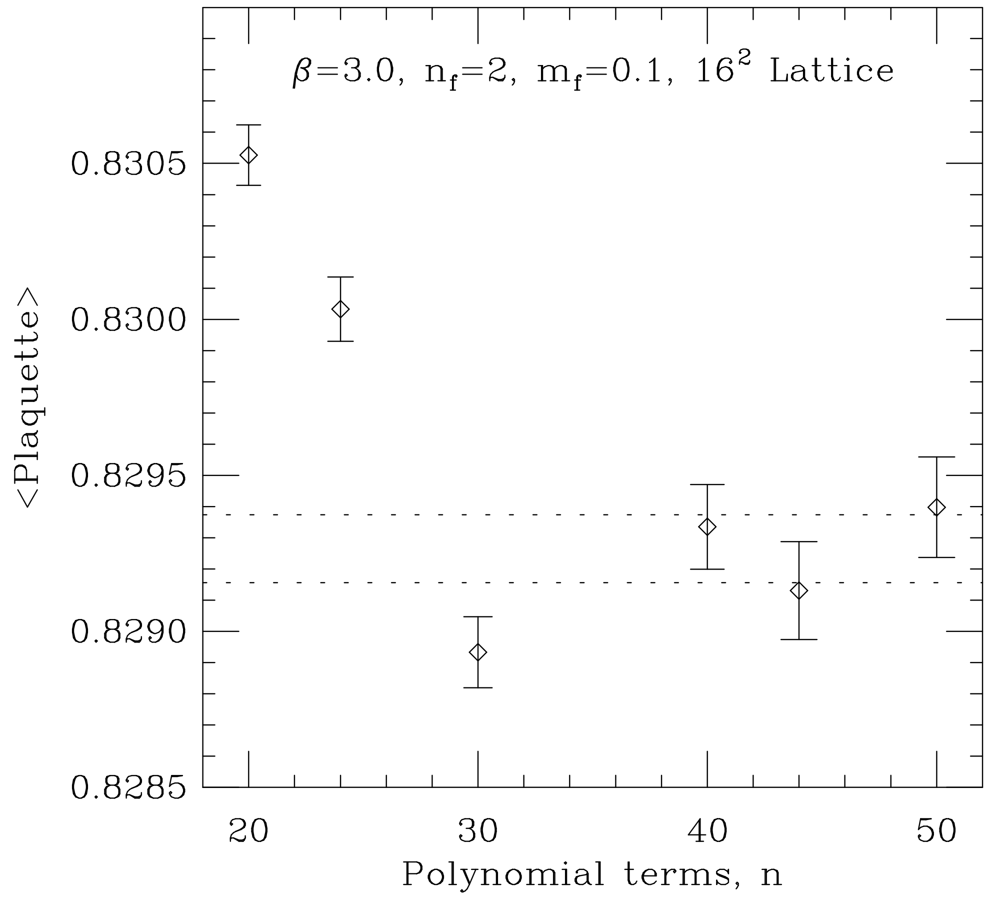


Figure 2.5: Plaquette expectation value *vs.* number of polynomial terms. Results are from simulations of the Schwinger model on a 16×16 lattice with $n_f = 2$, fermion mass $= 0.1$, $\beta = 3.0$. Dashed lines indicate the full theory result, calculated using the HMC algorithm.

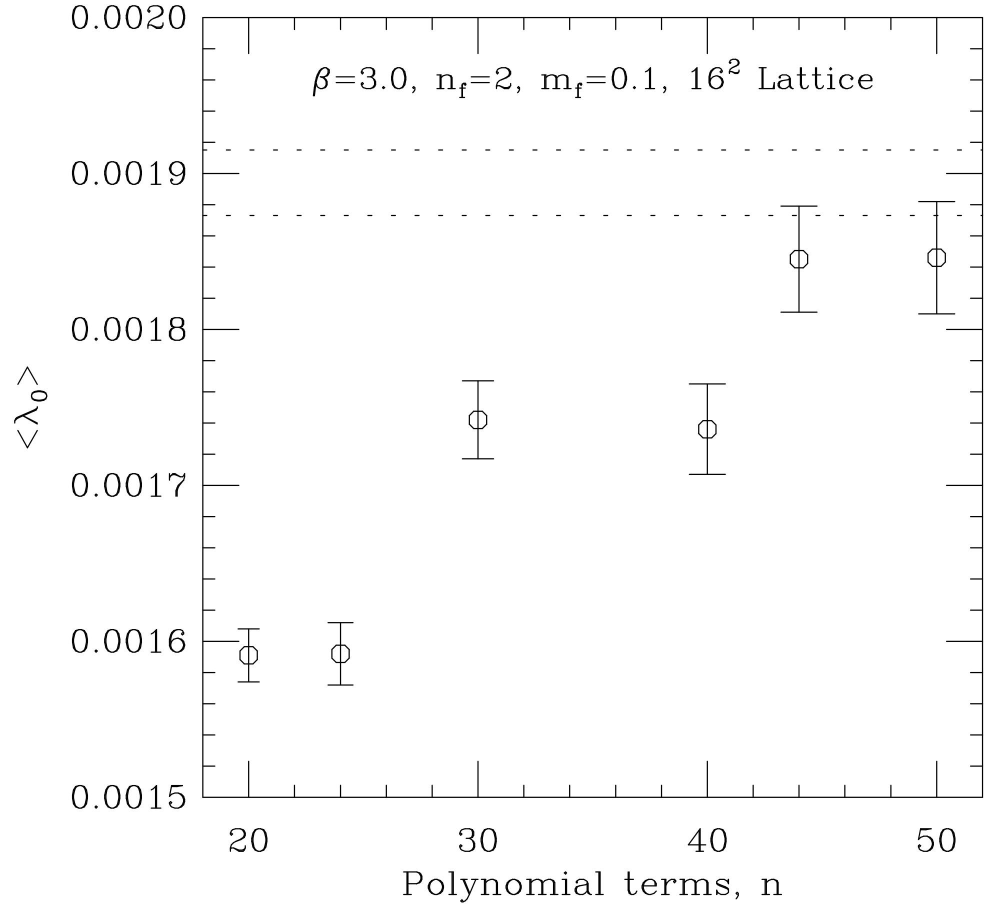


Figure 2.6: Expectation value of the smallest eigenvalue of $\not{D}$ *vs.* number of polynomial terms. Results are from simulations of the Schwinger model on a 16×16 lattice with $n_f = 2$, fermion mass = 0.1, $\beta = 3.0$. Dashed lines indicate the full theory result, calculated using the HMC algorithm.

2.6 Results - Autocorrelations

The true test of a Monte-Carlo algorithm's usefulness is the autocorrelation time. To study the performance of the method, Markov chains for $n = 30, 50, 60, 80$ and 100 polynomial terms were generated and the plaquette autocorrelations analysed.

For each parameter set, 7 Markov chains of 5000 configurations were constructed, and the plaquette measured on the ensemble. Each configuration was separated by one lattice-wide over-relaxation sweep of the gauge fields and one heatbath sweep over the set of auxiliary boson fields. The plaquette integrated autocorrelation times were then measured for each chain and averaged over the set of chains. For all cases, the resulting autocorrelation time was less than 10 times smaller than the ensemble length (5000 configurations).

The integrated plaquette autocorrelation time is presented for simulations with varying numbers of polynomial terms in figure 2.7. The Chebyshev parameter was fixed at $\epsilon = 0.005$. The dashed line on the figure is a straight line fit to the data points. This clearly demonstrates the result

$$\tau_{\text{auto}} \propto n. \tag{2.43}$$

This linear rise in the autocorrelation time has been observed in other gauge theory simulations [29] and is an important constraint on the performance of the algorithm (as will be discussed later). Figure 2.8 shows the variation of the autocorrelation time with the Chebshev parameter for the $n = 80$ polynomial. The dashed line is a fit to the form

$$\tau_{\text{auto}} \propto \frac{1}{\epsilon}. \tag{2.44}$$

Other groups have seen a rise in the autocorrelation time with the inverse square root of the Chebyshev parameter, however their runs were on small QCD simulations (lattice size 4^4) with heavy fermions.

Combining these two results gives the predicted behaviour of the algorithm as

$$\tau_{\text{auto}} \propto \frac{n}{\epsilon}. \tag{2.45}$$

2.6.1 Gauge Field Dynamics

Jegerlehner [29] provided an explanation for the linear rise in autocorrelation time, a greatly simplified version of his argument goes as follows. An estimate of the size of the effective interaction term from (2.40) gives

$$\langle \beta_{\text{eff}}(x) \rangle \propto \sum_{f=1}^{n} \langle \phi_f^* \phi_f \rangle, \tag{2.46}$$

and since the boson fields are nearly independent variables,

$$\langle \beta_{\text{eff}}(x) \rangle \propto n. \tag{2.47}$$

Hence for $\beta_{\text{eff}}(x)$ large, $\theta_\mu(x) - \Delta(x)$ is expected to be small and nearly gaussian distributed, so

$$\theta_\mu(x) - \Delta(x) \propto \frac{1}{\sqrt{\beta_{\text{eff}}(x)}} \propto \frac{1}{\sqrt{n}}. \tag{2.48}$$

The update of the single link variable will thus move θ by a small amount proportional to the inverse square root of n. To the link, a random walk of length $O(1)$ must be performed, which thus requires a number of updates proportional to n; hence

$$\tau_{\text{auto}} \propto n. \tag{2.49}$$

As well as this result, (2.46) gives information about the ϵ dependence of the autocorrelation time. The boson coupling matrix is $(\tilde{Q} - \{\text{Re } y_k\})^2 - \{\text{Im } y_k\}^2$ and hence, with reference to (2.7) the expectation value of the boson condensate grows in proportion to the inverse of $\sqrt{\epsilon}$. By the same random walk argument, the prediction is

$$\tau_{\text{auto}} \propto \frac{1}{\sqrt{\epsilon}}. \tag{2.50}$$

2.6.2 Auxiliary Boson Dynamics

The auxiliary boson fields are updated locally and will thus have autocorrelations too. Here, an argument for their influence on the autocorrelation of gauge field observables is presented by considering the simple dynamics of the bosons. The action for a single field is

$$S_f = \phi_f^* \left[(\tilde{Q} - \{\text{Re } y_k\})^2 - \{\text{Im } y_k\}^2 \right] \phi_f. \tag{2.51}$$

The matrix $\tilde{Q} - \text{Re}y_k$ will have a mode close to zero and can be regarded as an effective hopping term, hence the effective mass of the field is $m_k \equiv \text{Im } y_k$. So the field will have a correlation length of

$$\xi_k \propto \frac{1}{m_k}. \tag{2.52}$$

If the boson update algorithm has a scaling exponent of around 2, which is the case for the gaussian heatbath algorithm, then the autocorrelations of the boson field will be proportional to the effective mass squared. Hence, for a polynomial approximation with Chebyshev parameter, ϵ, the autocorrelations will scale as

$$\tau_{\text{boson}} \propto \frac{1}{\sqrt{\epsilon}}. \tag{2.53}$$

It is unclear which of the two different autocorrelation dynamics will dominate and govern the algorithm's behaviour. The prediction from the previous section for the gauge dynamics is

$$\tau_{\text{gauge}} \propto \frac{n}{\sqrt{\epsilon}}. \tag{2.54}$$

The results from section 2.6 suggest the two dynamics influence the performance simultaneously since the data suggests autocorrelations scaling like

$$\tau \propto \frac{n}{\epsilon}. \tag{2.55}$$

It must be emphasised that the data from section 2.6 are at only one set of physics parameters and a more involved study should be carried out to investigate the autocorrelations of the gauge and boson fields individually.

2.7 Autocorrelation Performance with Physical Parameters

The performance results of sections 2.6.1 and 2.6.2 agree with the results from section 2.6 in their prediction of the behaviour of the algorithm, given in (2.55). The free field study of section 2.3 links the accuracy of the approximation with the physical input parameters. Combining these two results, and assuming that keeping the error function of (2.30) constant at different fermion masses gives the same error in Monte-Carlo expectation values (which may not be the case and can not be tested reliably with the data available here) gives a prediction for the performance of the algorithm as the bare fermion mass is scaled of

$$\tau \propto \frac{1}{m_f^3}, \tag{2.56}$$

where the bare fermion mass is in lattice units and the Chebshev parameter is assumed to run with n to keep the polynomial error, δ of (2.6) constant. The boson field updates dominate the CPU cost of the method, and so the computer time required to generate an independent configuration is

$$\tau_{\mathrm{CPU}} \propto n\tau. \tag{2.57}$$

Thus the algorithm will scale as

$$\tau_{\mathrm{CPU}} \propto \frac{1}{m_f^4}. \tag{2.58}$$

This implies the algorithm behaves worse in the approach to the chiral limit than HMC, with predicted scaling of $1/m_q^{-13/4}$. The algorithm may be improved by a variety of methods and this point is discussed in section 2.8.

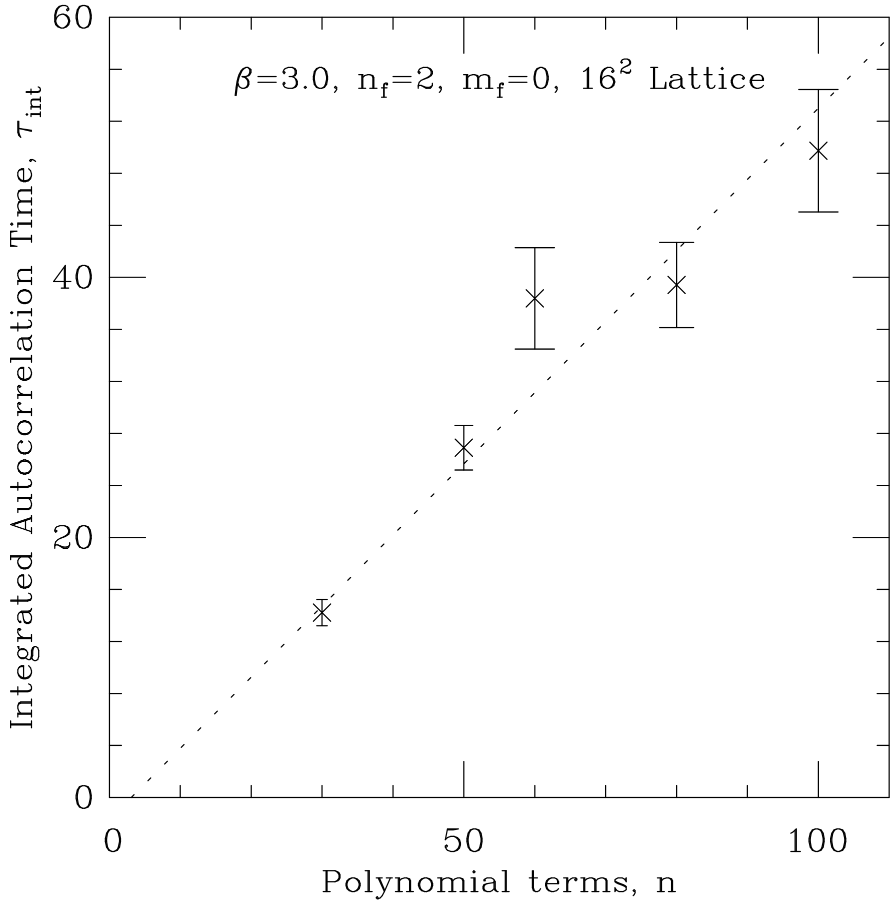


Figure 2.7: The plaquette integrated autocorrelation function *vs.* number of polynomial terms. Results are from simulations of the Schwinger model on a 16×16 lattice with $n_f = 2$, fermion mass = 0, $\beta = 3.0$. The Chebyshev parameter is held fixed at $\epsilon = 0.005$.

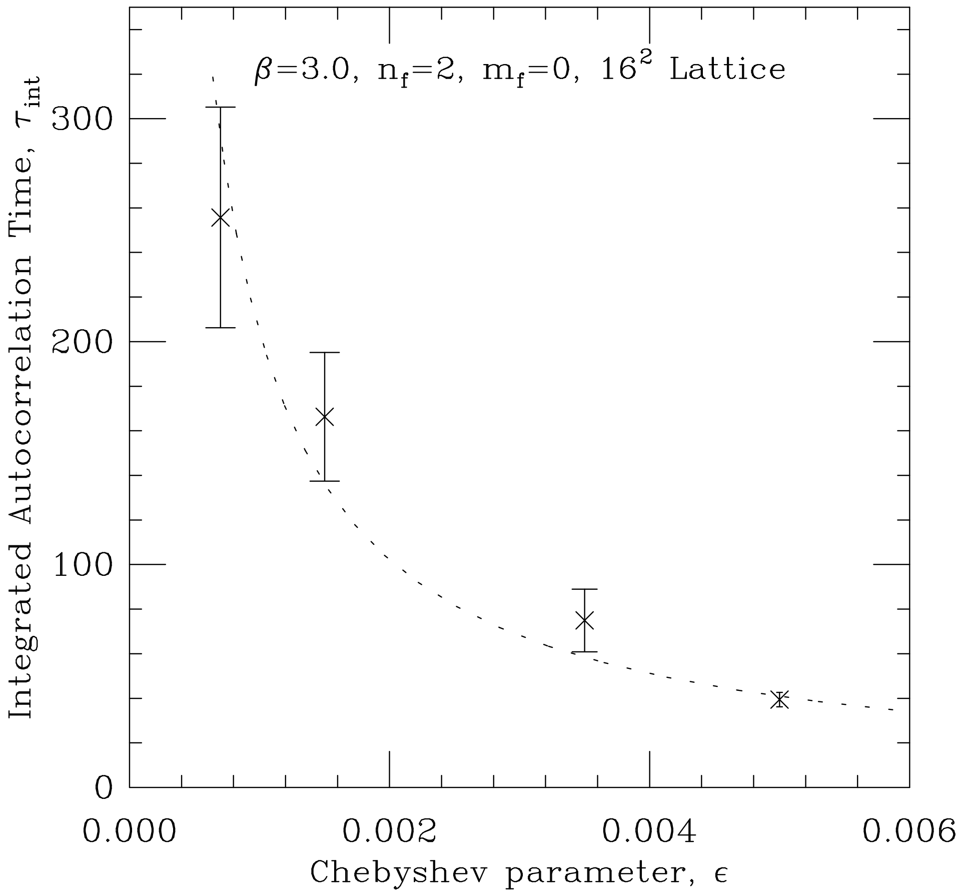


Figure 2.8: The plaquette integrated autocorrelation function *vs.* ϵ. Results are from simulations of the Schwinger model on a 16×16 lattice with $n_f = 2$, fermion mass $= 0$, $\beta = 3.0$. The algorithm uses an 80^{th} order polynomial approximation.

2.8 Conclusion

Lüscher's algorithm provides a promising base for development of dynamical fermion algorithms; the local interactions it generates make a wide variety of efficient Monte-Carlo updates such as over-relaxation applicable for the first time to this field. However this study has shown the method to suffer from some fundamental drawbacks.

The accuracy of the algorithm has been shown to be good (as expected given the exponential convergence properties of Chebyshev polynomials), for simulations of the Schwinger model, confirming studies performed on other theories (The Hubbard model, [30], $4d$ SU(2) [28] and QCD [31]). Convergence is good even for modest numbers of auxiliary boson flavours and with light fermion masses, however, the approach to the "true" unquenched theory is not monotonic in the number of polynomial terms employed. This implies an extrapolation from small n to $n \to \infty$ is fundamentally unreliable. In practice, however, results with statistical errors smaller than the systematic errors induced by the approximation are accessible even for light dynamical fermion masses (in this chapter, $m = 0.1$ was used).

The algorithm has been shown to have a linear rise in autocorrelation time with the number of boson fields used. This limit is the exact limit of the approximation and combining this effect with the difficulties in defining extrapolations to $n \to \infty$ makes use of the method difficult.

The investigation has suggested an autocorrelation time for the algorithm which scales like n/ϵ and has influences both from the boson and gauge field Markov dynamics. This result is based on rather unclear data and could be investigated more thoroughly by study of the gauge and boson fields individually. The boson field updates may be improved by incorporating over-relaxation techniques and it has also been suggested that the cluster algorithm could be employed (reducing the scaling exponent from 2 to 1) to reduce boson autocorrelations. The technique is still in its infancy and new approaches are still under investigation. Lüscher *et al.* proposed an alternative update scheme whereby the boson and gauge fields are updated simultaneously to attempt to sidestep the autocorrelation problem.

A possible extension to the method exists, whereby the valuable properties of the method, the simple nature of the local boson interactions and the exponentially convergent polynomial structure, are preserved while the systematic effects of the approximation are removed. This is discussed in chapter 3.

Chapter 3

LARD - an Exact Fermion Algorithm

An extension to the algorithm of chapter 2 is presented which, at the cost of an additional global Metropolis accept/reject step in the update process, ensures the equilibrium ensemble generated by the algorithm matches that of the fermionic partition function for any choice of polynomial. The addition of the Metropolis test means polynomial approximations with fewer terms can be employed, reducing the large autocorrelation times found when using the methods of the previous chapter, which require, for light fermion simulations, high order polynomial approximations to ensure reliable predictions.

The extension employs a similar philosophy to the HMC algorithm. Lüscher's method is used in analogy to molecular dynamics to provide an approximate update of the lattice configuration. For molecular dynamics, this update becomes exact in the limit of the integration stepsize, $\delta\tau \to 0$ and in Lüscher's method, the exact ensemble is recovered as the number of polynomial terms becomes infinite, *ie.* $1/n \to 0$. For both HMC and the technique discussed here, a Metropolis test is added which removes the approximation bias introduced when a finite stepsize or finite number of polynomial terms is used.

The extension is implemented in two alternative ways; one uses Lanczos diagonalisation to calculate the correction term required and the second uses the bosonic path integral to correct for polynomial errors stochastically.

Some performance results for the new algorithm applied to the Schwinger model are given, which show the algorithm outperforming HMC for one set of physical parameters by a factor of 5.

3.1 An Exact Dynamical Fermion Algorithm

The LARD (Local Action + Reduced Determinant) algorithm [32] exactly reproduces the dynamical fermion partition function by using the local boson update scheme but correcting for the presence of finite polynomial approximation errors with a weighting factor, whose influence on the Markov chain is exerted via a Metropolis accept/reject test.

3.1.1 Correcting the local boson partition function.

The partition function of a lattice gauge theory with dynamical fermions is, after trivially introducing a polynomial on $\tilde{Q}^2$

$$\begin{aligned} Z &= \int \mathcal{D}U \;\det\, \tilde{Q}^2 e^{-S_G[U]} \\ &= \int \mathcal{D}U \,\frac{\det \tilde{Q}^2 \det \mathcal{P}(\tilde{Q}^2)}{\det \mathcal{P}(\tilde{Q}^2)}\, e^{-S_G[U]}. \end{aligned} \tag{3.1}$$

If the polynomial is even with roots in complex conjugate pairs, then using the local boson partition function developed as an approximation to the full theory gives

$$Z = \int \mathcal{D}U \;\det\, \tilde{Q}^2\mathcal{P}(\tilde{Q}^2)\; Z_{\mathrm{LA}}(U)\; e^{-S_G[U]}. \tag{3.2}$$

Lüscher's fermion method is recovered by "quenching" (removing effects of its fluctuation over the ensemble from the partition function) the remaining non-local determinant in (3.2). In the exact limit of the polynomial approximation, the matrix product in the determinant will become 1 and hence the full theory and Lüscher theory are identical in spite of the "reduced quenching" effect. Now define a correction operator (the reduced determinant in (3.2)) as

$$\mathcal{O}[U] = \det\, \tilde{Q}^2\mathcal{P}(\tilde{Q}^2). \tag{3.3}$$

Since $\mathcal{P}(\tilde{Q}^2)$ is a polynomial on $\tilde{Q}^2$, the two matrices can be simultaneously diagonalised and so the operator of (3.3) can be calculated on a given gauge configuration by diagonalising $\tilde{Q}^2$ into its eigenvalues, $\{\tilde{\lambda}_i\}$. Then

$$\mathcal{O} = \prod_i^V \tilde{\lambda}_i \mathcal{P}(\tilde{\lambda}_i). \tag{3.4}$$

3.1.2 An Exact Algorithm

One method of correcting for the fluctuations in the operator of equation (3.3) is to reweight the contribution of configurations generated by the Lüscher method in the Monte-Carlo calculation of expectation values. So

$$\langle X \rangle_{\text{full}} = \frac{\langle X\mathcal{O} \rangle_n}{\langle \mathcal{O} \rangle_n}, \tag{3.5}$$

where $\langle \ \rangle_n$ is an average over the configuration ensemble generated by the Lüscher method with a polynomial of degree n. This technique will work well for good approximations, where the fluctuations in $\mathcal{O}$ are small and for expectation values where the correlation between the correction operator and the observable, $\langle X\mathcal{O} \rangle_n - \langle X \rangle_n \langle \mathcal{O} \rangle_n$, is small. The previous chapter provided results to indicate that the autocorrelation time of the local boson method rises linearly in the number of terms in the approximation, and so it would be advantageous to simulate with as small a number of terms as possible. Here, the fluctuations in $\mathcal{O}$ may be large.

An alternative is alter the Markov chain to include directly the effects of the correction operator. For any well-chosen polynomial, fluctuations in $\mathcal{O}$ will be significantly smaller than those in the determinant of the fermion matrix. The Lüscher method thus provides us with a scheme to propose good candidate configurations to a global Metropolis accept/reject step which will correct for the bias of $\mathcal{O}$.

3.1.3 Building the Exact Markov Chain

The computational cost of recalculating the correction operator upon updating a single link is prohibitively high so a large number of lattice-wide sweeps must be performed to cover the cost of the diagonalisation. The Lüscher scheme is built of update steps that change a single lattice link at a time and obey detailed balance for the approximate partition function of (2.21).

$$\begin{gathered} e^{-S_G[U]-S_{LA}[U,\phi_f]} \; R_{LA}(U \to U', \phi_f \to \phi'_f) \\ = \\ e^{-S_G[U']-S_{LA}[U',\phi'_f]} \; R_{LA}(U' \to U, \phi'_f \to \phi_f). \end{gathered} \tag{3.6}$$

A sequence of these single-link hits, whilst having the correct fixed point ensemble for the theory, does not necessarily obey the condition of (3.6). Detailed balance will be required if a Metropolis step is to be included. A chain of local (Lüscher action) updates will obey detailed balance if it is built reversibly. This imposes a constraint on the way the local boson method is implemented, but this imposition can be easily handled; this construction is discussed later. A reversible chain which sweeps the entire lattice any number of times can be built and so a significant step through phase space can be made between the expensive diagonalisation calculations.

With a chain of updates that obeys detailed balance for the Lüscher algorithm, add a Metropolis test on the correction operator, $\mathcal{O}[U]$. The transition probability for the full update (Lüscher + global Metropolis) for both the gauge and boson fields is

$$R_{\text{full}}(U \to U', \phi_f \to \phi'_f) = \min\left\{1, \frac{\mathcal{O}[U']}{\mathcal{O}[U]}\right\} R_{LA}(U \to U', \phi_f \to \phi'_f). \qquad (3.7)$$

With (3.6)

$$\begin{aligned} R_{\text{full}}(U \to U', \phi_f \to \phi'_f)\; e^{-S_G[U]-S_{LA}[U,\phi_f]}\; \mathcal{O}[U] \\ = \\ R_{\text{full}}(U' \to U, \phi'_f \to \phi_f)\; e^{-S_G[U']-S_{LA}[U',\phi'_f]}\; \mathcal{O}[U']. \end{aligned} \qquad (3.8)$$

The rate restricted to the gauge fields only is

$$R_{\text{full}}(U \to U') = \int \mathcal{D}\phi_f \mathcal{D}\, \phi'_f P(\phi_f|U)\; R_{\text{full}}(U \to U', \phi_f \to \phi'_f), \qquad (3.9)$$

where $P(\phi_f|U)$ is the probability of finding the auxiliary boson configuration $\{\phi_f\}$ on a given background gauge field configuration, U

$$P(\phi_f|U) \propto e^{-S_{LA}[U,\phi_f]}. \qquad (3.10)$$

Integration of the auxiliary boson fields in (3.8) and using the definition of $\mathcal{O}[U]$, (3.3) gives detailed balance for the gauge field transition as

$$R_{\text{full}}(U \to U')\ \det\ Q^2[U]e^{-S_G[U]} = R_{\text{full}}(U' \to U)\ \det\ Q^2[U']e^{-S_G[U']}. \qquad (3.11)$$

which has the fixed point ensemble probability distribution of the full theory,

$$P_{\text{full}}[U] = \frac{1}{Z} \det Q^2[U] e^{-S_G[U]}. \tag{3.12}$$

For the Lüscher update sequence to have the correct detailed balance property, the chain of single-site updates must be assembled reversibly. One simple method is to break the lattice down into a set of sublattices labelled by their even/odd parity, p and direction, μ. All the links on such a sub-lattice can be updated simultaneously, since there are no direct interactions between links on the same sublattice (see figure 2.4). The choice of order of updating the sublattices is made by randomly shuffling the 2d set of (μ, p) combinations. For the bosons, a similar idea applies. A $2d$ lattice is broken into 8 non-interacting sectors and all the sites in one sector are updated simultaneously. Bosons on sites (0,0), (2,2), (4,0) and (0,4) are in the same sector and can be updated simultaneously. As with the gauge fields, the eight boson sectors are updated in random order to ensure reversibility.

The integration over the bosonic fields in (3.8) requires (3.10) to be true which implies the fields ϕ_f are in equilibrium with the gauge configuration, U. This must also hold for the right hand side of the equation *ie.* the boson fields ϕ'_f are in equilibrium with the configuration U'. Hence if the Metropolis test rejects, both the gauge and auxiliary boson configurations must be reset to their previous value. This is in spite of the fact that the final gauge field ensemble probability is independent of the auxiliary boson fields (the probability is that of the dynamical gauge theory and is, as such independent of the structure of the polynomial).

The correction operator $\mathcal{O}[U]$ can be reformulated for the $2^{\frac{d}{2}}$ fermion flavour case as discussed for the Lüscher method in section 2.1.3 by noting that the even-even sites and the odd-odd sites of the matrix $\tilde{Q}^2\mathcal{P}(\tilde{Q}^2)$ decouple [17]. Now the Lüscher update scheme is used with $n/2$ flavours of bosons and the Metropolis accept/reject decision is made after a calculation of

$$\sqrt{\det \tilde{Q}\mathcal{P}(\tilde{Q})} = \mathcal{O}_e[U] = \det{}_e \tilde{Q}^2_{ee}\mathcal{P}(\tilde{Q}^2_{ee}) = \prod_i^{V/2} \tilde{\lambda}_i \mathcal{P}(\tilde{\lambda}_i). \tag{3.13}$$

3.1.4 Summary of the Exact Algorithm

The algorithm is

- Calculate the correction term, $\mathcal{O}[U]$.
- Store the gauge and auxiliary boson field configuration, $\{U, \phi_f\}$.
- Perform j update steps of all the auxiliary boson fields and the gauge boson fields over the lattice using a reversible scheme. This scheme obeys detailed balance for Lüscher's method.

$$e^{-S_G[U]-S_{LA}[U,\phi_f]}P(U \to U', \phi_f \to \phi'_f)$$
$$=$$
$$e^{-S_G[U']-S_{LA}[U',\phi'_f]}P(U' \to U, \phi'_f \to \phi_f).$$

- Recalculate the correction, $\mathcal{O}[U']$ by diagonalising $\tilde{Q}^2$.
- Accept or reject the candidate configuration from the Lüscher scheme, $\{U', \phi'_f\}$ by performing a Metropolis test,

$$P_{\text{acc}} = \min\left\{1, \frac{\mathcal{O}[U']}{\mathcal{O}[U]}\right\}.$$

Since the algorithm is exact for any polynomial, any number of auxiliary boson fields can be used in the simulation. The quality of the polynomial approximation now governs the acceptance rate of the global Metropolis test. The natural way to exploit this gain is to avoid the high autocorrelations found for Lüscher's method with high order polynomials. This also reduces the large memory consumption for the method which introduces n boson fields. The cost is the need to calculate the influence of the reduced determinant of (3.3).

Every time a Metropolis accept/reject decision is made, $\mathcal{O}[U]$ must be recalculated on the current configuration. The calculation of the correction operator requires the diagonalisation of the fermion matrix. This is achieved by using the Lanczos algorithm (see section 3.4.1). This algorithm is computationally intensive and requires $V^2 \not{D}$ operations. For a $4d$ theory, such as QCD, the cost of the diagonalisation may be prohibitively high and grows rapidly as the lattice size increases [31].

3.2 An Alternative Implementation - Guide Bosons

In this section an alternative implementation of the global Metropolis test is presented which avoids the need to perform an expensive fermion matrix diagonalisation. Stochastic degrees of freedom, called "guide bosons" are introduced to compensate for fluctuations in the correction term. This field of guide bosons has a non-local action. The bosons are held fixed during the gauge field and auxiliary Lüscher boson field updates and the global Metropolis test is performed on the change in the "guide" action. After every Metropolis test the guide boson fields are globally refreshed from an exact gaussian heatbath. The technique is similar to an extension discussed in [33, 34, 35]

3.2.1 The Guide Boson Action

The correction operator, $\mathcal{O}[U]$ is

$$\mathcal{O}[U] = \det \tilde{Q}^2 \mathcal{P}(\tilde{Q}^2).$$

For an even order polynomial with roots in complex conjugate pairs, $\mathcal{P}(\tilde{Q}^2)$ is positive definite (and commutes with $\tilde{Q}$). Hence the bosonic path integral can be employed directly (as with the pseudofermions used in HMC simulations) to calculate the operator

$$\mathcal{O}[U] = \int \mathcal{D}\chi \mathcal{D}\chi^* e^{-\chi^* X^{-1} \chi}. \tag{3.14}$$

with

$$X = \tilde{Q}^2 \mathcal{P}(\tilde{Q}^2) = \tilde{Q}^2 \prod_k^{n/2} (\tilde{Q}^2 - z_k)(\tilde{Q}^2 - z_k^*) \propto Q^2 \prod_k^{n/2} (Q^2 - y_k)(Q^2 - y_k^*). \tag{3.15}$$

This new set of guide bosons has a non-local, positive definite action, S_χ given by

$$S_\chi = \chi^* X^{-1} \chi. \tag{3.16}$$

For good polynomial approximations, fluctuations in the action of (3.16) are small. In the exact limit, the matrix X becomes the identity and the guide bosons decouple from the gauge field.

An equilibrated ensemble of bosons can be generated stochastically. From (3.15), X can be written as $Y^\dagger Y$ with

$$Y = Q \prod_k^{n/2} (Q^2 - y_k), \tag{3.17}$$

and if a source of gaussian noise, η is generated, then the guide boson heatbath is

$$\chi = Y\eta. \tag{3.18}$$

The stochastic algorithm is then similar to the exact algorithm proposed in section 3.1.

3.2.2 Summary of the Guide Boson Algorithm

The alternative algorithm is

- Refresh the guide boson fields by generating a field of gaussian noise, η and calculate their action, $S_\chi[U] = \chi^* X^{-1} \chi = \eta^* \eta$
- Store the gauge and auxiliary boson field configuration, $\{U, \phi_f\}$.
- Perform j update steps of the auxiliary boson fields and gauge boson fields using a reversible scheme
- Recalculate the guide boson action on the new gauge configuration, $\{U'\}$ (by solving $\mu = Y^{-1}\chi$).
- Accept or reject the candidate configuration from the Lüscher scheme, $\{U', \phi'_f\}$ by performing a Metropolis test on the change in the guide boson action,
$$P_{\text{acc}} = \min\left\{1, e^{-\Delta S_\chi}\right\}.$$

3.2.3 Halving the Fermion Flavours with the Guide Action

If the number of fermion flavours is to be halved to $2^{\frac{d}{2}}$, then the natural even/odd decomposition of the matrix X can be exploited [17]. X has only even-even and odd-odd coupling. The guide bosons on one parity of the lattice can be removed to take the fermion determinant square root. The action (for bosons on one parity only) is now

$$S_e = \chi_e^* X_{ee}^{-1} \chi_e \tag{3.19}$$

The global heatbath becomes more complicated, since Y couples sites of different parity. First, a stochastic field, $\bar{\eta}_e$ existing on even parity only, is built from gaussian fields on both parities, (η_e, η_o) by

$$\bar{\eta}_e = \not{D}_{eo}\eta_o + m\eta_e. \tag{3.20}$$

Then the guide boson fields, with an ensemble generated by (3.19) are

$$\chi_e = \prod_k^{n/2} (Q_{ee}^2 - y_k)\bar{\eta}_e. \tag{3.21}$$

The action is calculated by inverting the matrix X_{ee} and then

$$S_e = (\chi_e^* X_{ee}^{-1}) X_{ee} (X_{ee}^{-1} \chi_e). \tag{3.22}$$

3.3 Tuning the Algorithm

The proof of exactness presented in section 3.1.3 imposes no constraint on the form of the polynomial $\mathcal{P}$. The method is exact for any choice of polynomial however use of Lüscher's auxiliary bosons and the stochastic correction method constrains the polynomial to be even and have roots in complex conjugate pairs. The discussion will be restricted to this class of polynomial. The number of terms in the polynomial is now a free parameter, unlike the uncorrected Lüscher method of chapter 2 where the polynomial approximation errors govern the reliability of results generated. The quality of the approximation to $1/s$ now controls the acceptance rate of the Metropolis test. The linear rise in autocorrelation time with the number of polynomial terms used for the Lüscher method (illustrated in chapter 2) means n should be set as low as possible while preserving a good acceptance rate for the Metropolis test. Since the acceptance rate is optimal for good approximations to the inverse, it seems natural to retain the exponential convergence properties of the Chebyshev polynomial analysis in section 2.1.1. The

Chebyshev scale parameter, ϵ is no longer restricted to be close to the smallest eigenvalue of $\tilde{Q}^2$ as was required for optimisation of the Lüscher algorithm and it becomes another tunable parameter governing the performance of the update scheme. This might allow the increase in autocorrelations with small values of this parameter to be avoided. It should also be noted that the matrix in the polynomial does not need to be the fermion matrix of the final ensemble. An equally valid starting point for the procedure of section 3.1 would be

$$Z = \int \mathcal{D}U \, \frac{\det \tilde{Q}^2 \det \mathcal{P}(\mathcal{Q}^2)}{\det \mathcal{P}(\mathcal{Q}^2)} \, e^{-S_G[U]}. \tag{3.23}$$

With $\mathcal{Q}$ any matrix which can be simultaneously diagonalised with the required fermion matrix of the simulation. The constraint of simultaneous diagonalisability is not strictly enforced but if $\mathcal{Q}$ can not be diagonalised with Q then two Lanczos diagonalisations must be performed. One choice is to shift the bare fermion mass in the matrix *ie.*

$$\mathcal{Q} = \tilde{Q}(m' \neq m) = \frac{\Gamma_5(\not{D} + m')}{(d^2 + m'^2)}. \tag{3.24}$$

Again, this shifted mass is a free parameter (the ensemble generated is independent of m') and can be tuned to optimise the autocorrelation performance of the algorithm. Tuning these parameters is similar to tuning the parameters of HMC, where it has been suggested [36, 37] that the finite stepsize errors in the molecular dynamics scheme renormalise the bare parameters and use of shifted masses may compensate for this effect. For example, tuning n is equivalent to tuning the stepsize, $\delta\tau$. Mass tuning was not tested in this study, but preliminary strong coupling analysis suggested only a marginal reduction in the fluctuations of the correction operator (leading to improved acceptance rates) with tuning. Since the auxiliary boson correlation lengths are governed by the polynomial roots, no hidden gain in speeding up the auxiliary boson dynamics seems likely.

Since a global Metropolis test has been added already, then for the extra cost of a calculation of the lattice gauge action, the update scheme could be performed at shifted gauge coupling, β', with the accept/reject test correcting for the shift. The shifted gauge coupling could be chosen to try to correct for the polynomial errors which could lead to better global acceptance rates. Again, testing this idea is beyond the scope of this thesis.

3.4 Implementing the Method for the Schwinger Model

The gauge configurations and auxiliary bosons are updated in essentially the same way as for the simulations of chapter 2, with reversibility added to ensure detailed balance. In order to calculate $\mathcal{O}[U]$, a Lanczos diagonalisation scheme is used.

3.4.1 Lanczos Diagonalisation

The Lanczos method (with explicit re-orthogonalisation) diagonalises a hermitian matrix completely by first constructing a tridiagonal basis of states and re-expressing the matrix in this basis. The tridiagonal form is then fully diagonalised using a standard technique (such as `tqli` [38]).

This works very well for the hermitised fermion matrices on lattices of up to 20×20. Beyond this size, significant rounding errors make the method increasingly unstable.

3.4.2 Inverting the Correction Matrix

The guide boson method has less problematic volume dependence, and can be inverted with very few iterations of a standard solver method such as conjugate-gradient. The performance of the solver acting on the matrix X_{ee} for simulations with the $2^{\frac{d}{2}}$ action introduced in section 3.2.3 is discussed in section 3.9.

For each Metropolis test, the guide boson action of (3.16) must be recalculated. This requires the inversion of Y for the $2^{\frac{d}{2}+1}$ flavours case, or X_{ee} for the $2^{\frac{d}{2}}$ flavours case. For $2^{\frac{d}{2}+1}$,

$$S_\chi = \chi^* X^{-1} \chi = \chi^* \left[Y^\dagger Y\right]^{-1} \chi. \tag{3.25}$$

Solving

$$\mu = \left[Y^\dagger\right]^{-1} \chi, \tag{3.26}$$

gives $S_\chi = \mu^* \mu$.

For a good polynomial approximation to the inverse, X is well conditioned. This implies Y will be well conditioned and standard iterative solver methods, like conjugate gradient, will converge very rapidly. Section 3.9 gives results related to the cost of the inversion of X_{ee} with a conjugate-gradient solver.

3.5 A Strong Coupling Analysis

Fluctuations of the error operator of (3.3) over an ensemble were studied in the strong coupling limit in order to gain some insight into their magnitude as the polynomial parameters, as well as the fermion mass are varied. This should give predictions as to the change in acceptance rate as the fermion mass of the theory varies. The variance of $\mathcal{O}$ on an ensemble is defined as

$$\mathrm{var}(\mathcal{O}) = \frac{\langle \mathcal{O}^2 \rangle - \langle \mathcal{O} \rangle^2}{\langle \mathcal{O}^2 \rangle}. \tag{3.27}$$

The variance of $\mathcal{O}$ on a set of 100 $\beta = 0$ configurations for three polynomials as a function of the Chebyshev parameter is shown in figure 3.1. The horizontal line gives the fluctuations in the determinant of $\tilde{Q}^2$ alone. As with the free field results of 2.3, the optimal value of ϵ is higher than the smallest eigenvalue of $\tilde{Q}^2$, implying the Metropolis test will have a higher acceptance rate when this parameter is tuned to be above this lowest eigenvalue. Note also that the optimised value approaches $\lambda_{\mathrm{min}}[\tilde{Q}]$ as the order of the polynomial grows.

In figure 3.2, the variance of the operator, $\mathcal{O}[U]$ (with ϵ tuned to minimise fluctuations) is shown as a function of the number of polynomial terms and fermion mass. The variance falls exponentially with both the fermion mass and number of polynomial terms as expected for the Chebyshev acceleration scheme. Hence

$$\log \mathrm{var}(\mathcal{O}) \propto -n m_f a. \tag{3.28}$$

This result then implies that for a constant acceptance rate as the fermion mass is reduced, the number of polynomial terms must rise in inverse proportion

$$n \propto \frac{1}{m_f} \quad \text{for constant Metropolis acceptance} \tag{3.29}$$

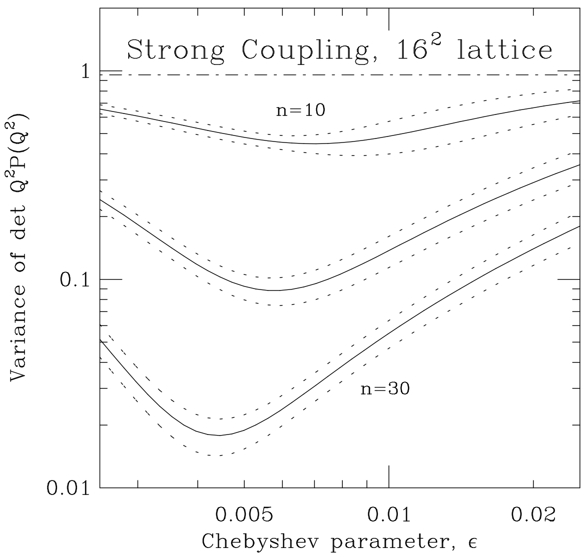


Figure 3.1: Fluctuations in the error function in the strong coupling limit . The lines are for $n = 10, 20$ and 30 polynomial terms. Results are for a 16×16 lattice, with fermion mass, $m = 0.1$. The upper line indicates the variance of the full fermion determinant on the same set of configurations.

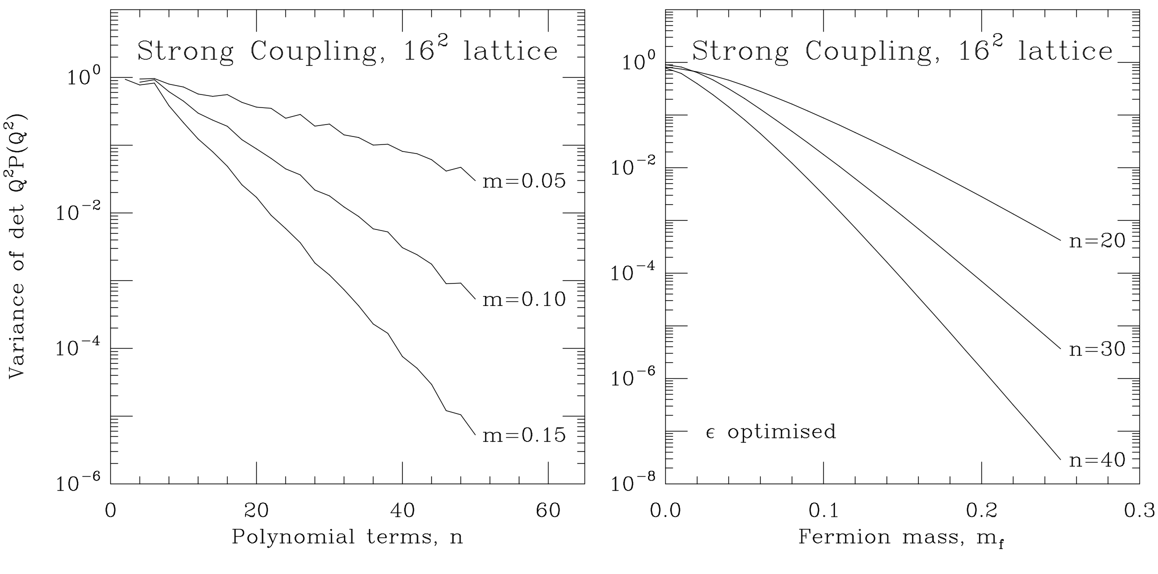


Figure 3.2: The dependence of the fluctuations in the strong coupling correction operator on the number of polynomial terms and the fermion mass.

3.6 Results - An Exact Algorithm

The Markov analysis of section 3.1.3 suggests the algorithm reproduces the exact dynamical fermion ensemble for any choice of polynomial. This prediction was tested by calculating the expectation value of Q^2_{top} (related to the topological susceptibility of the theory) for a poorly convergent polynomial using both Lüscher's method (which should show significant errors as the approximation breaks down) and the new algorithm incorporating the Metropolis scheme with a Lanczos correction. A simulation at $\beta = 3.0$ with two flavours of $m = 0.01$ dynamical fermions on a 16×16 lattice was performed using a polynomial of order 8. The simulation was run using a variety of values for the Chebyshev parameter. Results from this set of simulations are shown in figure 3.3. The horizontal line is the HMC algorithm's prediction for $\langle Q^2_{\mathrm{top}} \rangle$. Circles are results from the LARD algorithm, crosses indicate data from uncorrected Lüscher simulations. The graph demonstrates that, within statistical errors, the Metropolis test successfully removes the errors induced by the use of polynomial approximations, even when the Lüscher method shows significant deviations.

3.7 Results - Acceptance Rates

The exponential convergence of Chebyshev polynomial approximations should give an update scheme with a good acceptance rate, even for low order polynomials. The acceptance rates of the method using both the Lanczos and guide boson implementations were studied. Figure 3.4 shows the acceptance for simulations using low order polynomials ($n = 8, 10$ and 20) with light dynamical fermions (two flavours, $m_f = 0.01$). Even for a polynomial of degree 8, the optimised acceptance probability is as high as 37%. The optimisation results are given in table 3.1.

Polynomial terms, n	8	10	20
ϵ_{opt}	0.0075	0.0075	0.0050
$P_{\mathrm{acc}}(\epsilon_{\mathrm{opt}})$	0.37(1)	0.44(1)	0.63(1)

Table 3.1: Optimised Acceptance rates. $\beta = 3.0$, $m_f = 0.01$, 16×16 lattice.

With the introduction of guide bosons, more stochastic noise has been added to the system and accordingly, the fluctuations in the correcting term used in the Metropolis are larger than for the Lanczos method. This leads to a lower

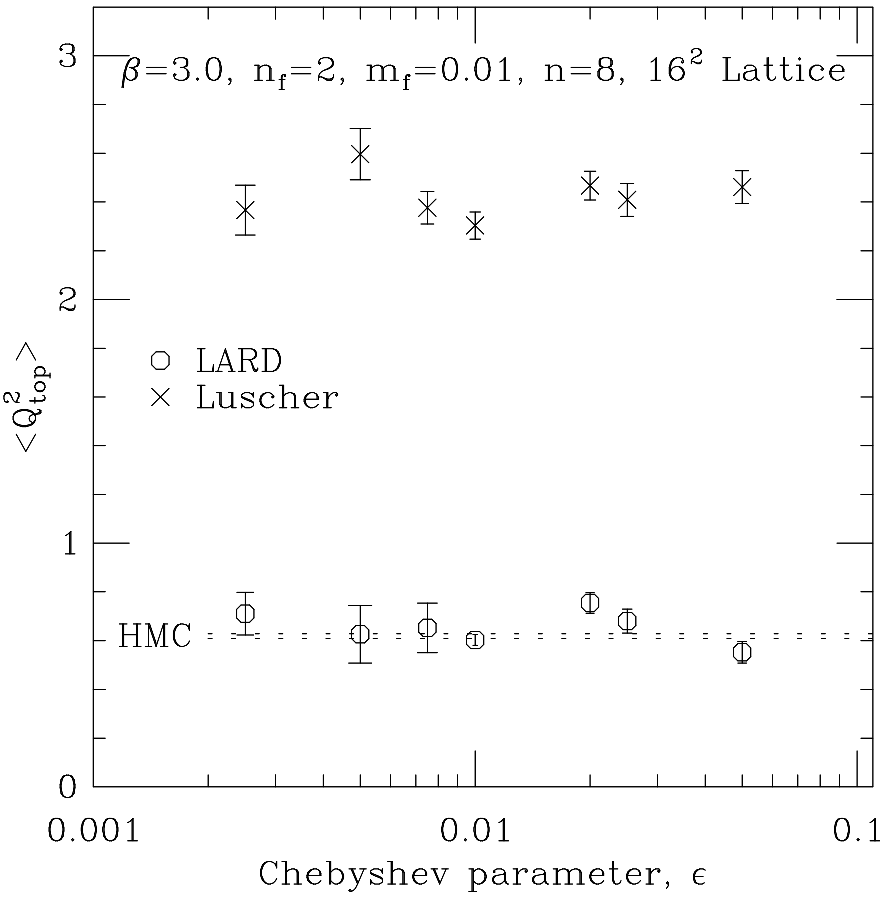


Figure 3.3: A test of exactness after including the Metropolis test. Results are from simulations of the Schwinger model on a 16×16 lattice with $n_f = 2$, fermion mass $= 0.01$, $\beta = 3.0$. Crosses ($\times$) indicated results generated without the Metropolis correction using a $n = 8$ polynomial. Circles ($\circ$) indicate the results using the same polynomial approximations after correction. The dashed lines indicate the HMC result.

acceptance rate for this method. This result is demonstrated in a simulation using the two methods with identical physical and Chebyshev polynomial parameters. Figure 3.5 shows the acceptance probability with the two schemes for a simulation on a 16×16 lattice, with $\beta = 3.0, m_f = 0.1$ and a 20^{th} order polynomial. Note that both methods have a maximum acceptance rate at the same value of the Chebyshev parameter.

3.8 Results - Autocorrelations; Comparing with HMC

An attempt to assess whether the exact algorithm with the Lanczos correction can outperform HMC, currently the most popular dynamical fermion algorithm for even numbers of fermion flavours was made. The study also tried to find the optimal choice of polynomial for simulations with the Metropolis test.

The test was performed using a range of polynomials of order $n = 8$, 10 and 20 at $\beta = 3.0$, $m_f = 0.01$ on a 16×16 lattice. For each parameter set, three Markov chains of 3000 sweeps were generated (for all the simulations, where all the integrated autocorrelation times are less than 100 sweeps, this length of chain should give a reliable estimate of autocorrelations). The topological charge was measured on the chains and autocorrelations measured in this observable. Q_{top} has the advantage for autocorrelation measurements of having (by the charge conjugate symmetry of the action) zero expectation value in all simulations.

The results are given in figure 3.6. The autocorrelation times at each of the three polynomial orders, n are scaled by n to include the extra CPU cost of simulating with an increasing number of boson fields. This should give a true performance comparison. The results indicate that performance of the algorithm seems to be optimised for lower numbers of auxiliary boson fields, where the update dynamics avoid the autocorrelation problems of chapter 2, whilst still maintaining a good acceptance rate. Simulations below $n = 8$ are hampered by very low Metropolis acceptance rates. The Chebyshev parameter favoured by the algorithm is high too, and indicates the algorithm favours the regime which avoids the increase in autocorrelation times found as ϵ is reduced.

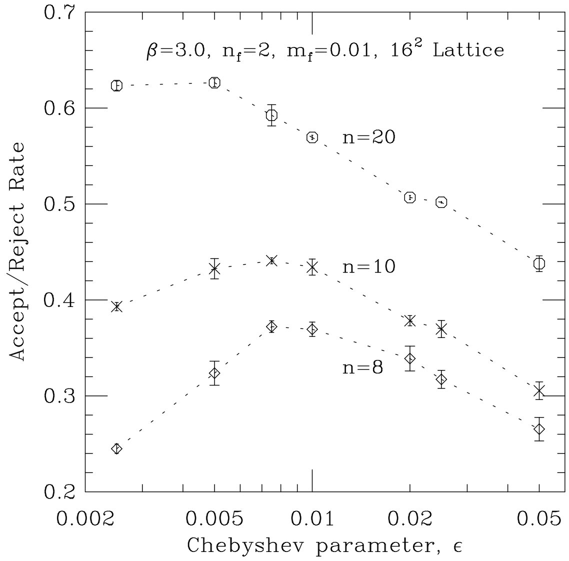


Figure 3.4: Acceptance probabilities for the global Metropolis step of the LARD algorithm. The three sets of data correspond to $n = 8(\diamond), 10(\times)$ and $20(\circ)$ respectively. Dashed lines are to guide the eye.

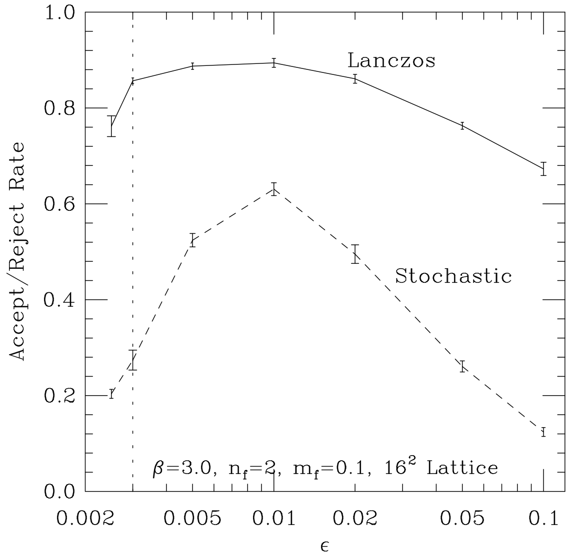


Figure 3.5: Comparing acceptance probabilities for the global Metropolis step of the LARD algorithm using Lanczos and stochastic (guide boson) corrections.

	LARD $n = 8$	LARD $n = 20$	HMC $\delta\tau = 0.05$
Update time	1.8 secs	3.7 secs	6.7 secs
$\tau_{\rm int}$(sweeps)	8 ± 1	13 ± 3	13 ± 1
$\tau_{\rm CPU}$	14 ± 2	48 ± 10	87 ± 7

Table 3.2: Performance comparison of HMC *vs.* LARD for the Schwinger model. The timings are from simulations on a DEC Alpha Workstation (`alsace.ph.ed.ac.uk`)

Timings for the alternative updates are given in table 3.2. For the HMC algorithm, the time is for one trajectory of unit simulation time, consisting of 20 leapfrog steps of $\delta\tau = 0.05$ (chosen to give an acceptance rate of $\approx 70\%$ which is optimal). The LARD results are timings for $3 \times (1 \times$ gauge over-relaxation sweep $+ 3 \times$ boson pseudo-heatbath sweeps). From these results, a clear improvement over HMC can be seen for both LARD runs. The $n = 8$ run is about 5 times faster than HMC for this light fermion simulation.

3.9 The Guide Boson Solver

To assess the overhead added by the algorithm when the guide boson correction scheme is employed, a study of the conditioning number of the coupling matrix, $X_{\rm ee}$ was made. The conditioning number of a matrix, κ is defined as the ratio of the largest and smallest eigenvalues of the matrix

$$\kappa = \frac{\lambda_{\rm max}}{\lambda_{\rm min}}. \tag{3.30}$$

This gives an indication of the number of conjugate-gradient (or other iterative solver) sweeps required for inversion. Since the matrix X involves $2n$ $\not{D}$ operations, a measure of the CPU time involved in inverting the matrix X is $n\kappa$. After optimising the Chebyshev parameter, the effective conditioning number becomes weakly dependent on n and thus the cost of carrying out the guide boson Metropolis test is independent of n.

Figure 3.7 illustrates the number of solver iterations required to invert X_{ee} to a residue of $r^2 = 10^{-18}$. Again, tuning the Chebyshev parameter is a crucial step in optimisation. For the same set of configurations, a conjugate gradient inversion of Q^2_{ee} required 76.2 ± 0.6 iterations. For $n = 20$ the number of $\not{D}$ operations performed to invert X_{ee} for optimal ϵ was 164 ± 2 and so calculating the guide boson action takes between 2-3 times longer than inverting the fermion matrix. The Metropolis test then has a CPU cost similar to the HMC Metropolis test.

3.10 Conclusions

The LARD algorithm has been proposed as an extension to the update scheme generated by the local boson action discussed in chapter 2. The addition of a Metropolis acceptance test after a number of local boson action sweeps removes the bias in the original algorithm induced by the use of polynomial approximations. Two alternative implementations have been discussed. One uses the Lanczos method to exactly diagonalise the fermion matrix to assess the required correction, the other uses a stochastic estimator. This second implementation is more suited to $4d$ gauge theories (such as QCD) since it avoids the costly diagonalisation step (which has an overhead that grows like V^2). The stochastic implementation has however been shown to have a lower acceptance rate for the Metropolis step. An estimate by Boriçi and de Forcrand [33], based on the increase in number of solver iterations with volume suggests the cost of the inversion will only grow as $V(\log V)^2$.

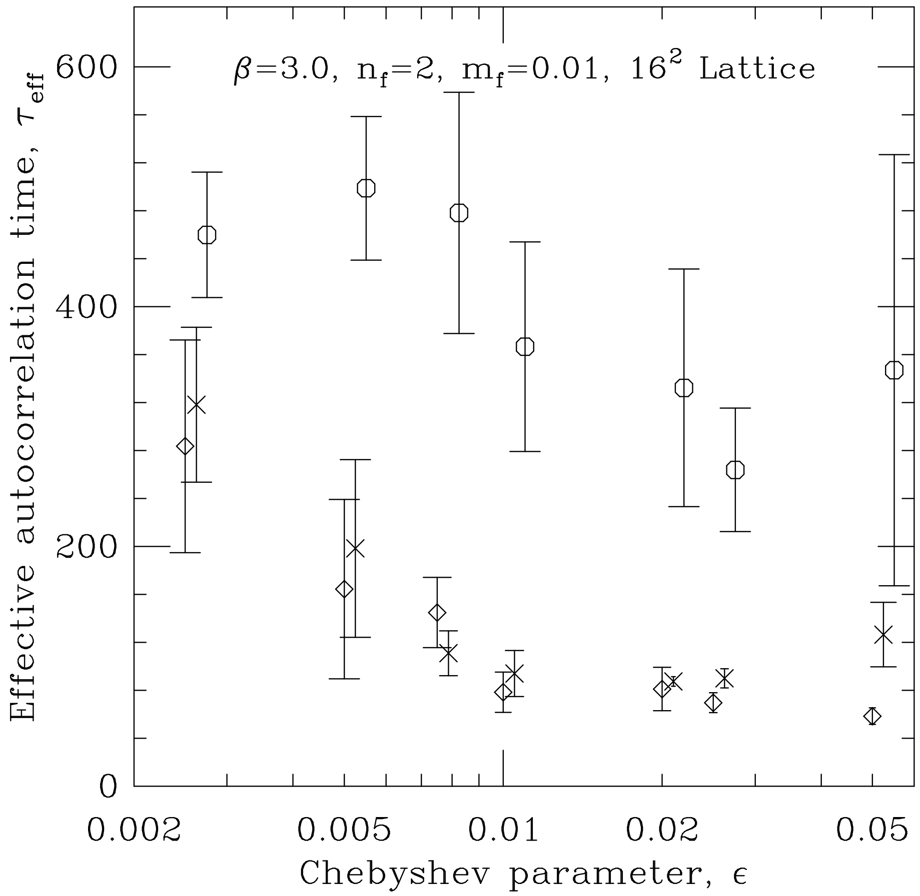


Figure 3.6: Autocorrelations for the LARD algorithm. The integrated autocorrelation time is measured for the topological charge and scaled by the number of auxiliary boson fields to give a measure of the autocorrelations in CPU time. Data are for three different orders of polynomial, $n = 8(\diamond), 10(\times)$ and $20(\circ)$.

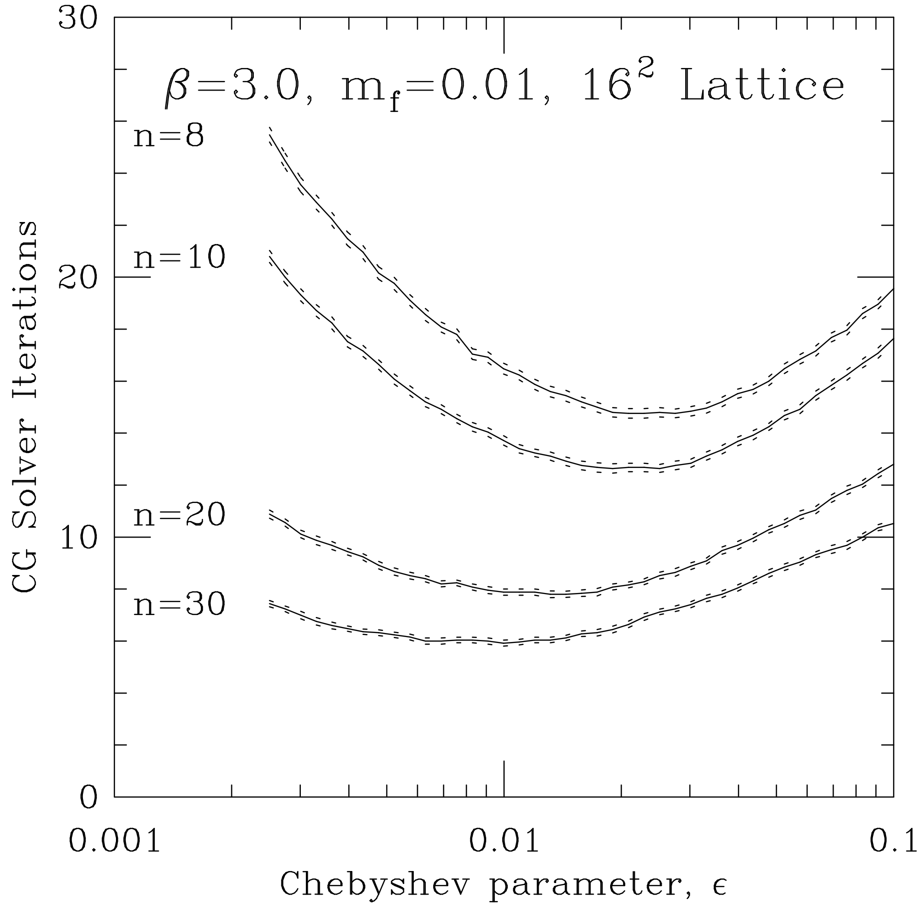


Figure 3.7: Conjugate-gradient iterations required to invert X_{ee}. Results are from 25 16×16 dynamical configurations, $\beta = 3.0$, $m_f = 0.01$. The target residue for convergence was fixed at $r^2 = 10^{-18}$

Strong coupling results indicate that, as with the original algorithm, the number of polynomial terms required will grow in inverse proportion to the fermion mass. If this is the case and if the Markov dynamics of the gauge and auxiliary boson fields are the same, then the algorithm will have the same scaling as the uncorrected Lüscher method.

With the exact algorithm, the Chebyshev parameter has taken on a new role; that of a free, tunable control which can be maximised to increase the efficiency of the algorithm. It remains unclear how the value of this parameter changes as the fermion mass changes and so the algorithm's scaling properties can not be reliably estimated.

Using the algorithm with a Lanczos diagonalisation, dynamical simulations have been carried out about 5 times faster than HMC for one set of physical parameters. The tuning of the algorithm has been discussed but, as with simulating with HMC an extensive "folklore" for optimisation will need to be developed. The scaling behaviour of the new method remains unclear.

Chapter 4

Testing improved QCD lattice actions; The Glueball Spectrum

In this chapter, a Symanzik-improved lattice action [39] , designed to remove the $O(a^2)$ errors in the Wilson action and incorporating the tadpole improvement scheme [40, 41] is tested by calculating the masses of the glueball spectrum using large lattice spacings. Improved scaling behaviour is demonstrated for the scalar (0^{++}) glueball but no reliable signal is found for other glueball states. The test exposes two problems with using such an action. The coarse lattice spacing means that correlators of the heavy glueballs fall rapidly and the Monte-Carlo signal is lost in vacuum fluctuations after a few timeslices. The standard method for curing this condition is to employ a variational technique [42] whereby a large number of operators is used to create glueball states with the appropriate quantum numbers and the decay of an optimal correlator, built from a linear combination of the set of creation operators is examined. This method should give a reliable correlator plateau at small source-sink separations, where the signal is still reliable. Unfortunately for the Symanzik-improved action with terms coupling gauge links separated by two time steps, the transfer matrix is no longer hermitian, and the variational calculation becomes unreliable.

4.1 The Symanzik-Improved Gauge Action

As discussed in Chapter 1, the computational cost of a lattice Monte-Carlo calculation rises rapidly as the continuum limit is approached. Naturally then, it seems advantageous to simulate as far from this limit as possible. The constraint on the maximum lattice spacing that can be employed is twofold. Firstly, the grid must be fine enough so that physical objects of interest stretch over a few lattice points. A rule of thumb employed in a wide range of discretisation applications is that the lattice size of a typical scale in the object of interest is about three discrete grid points across. As an example, for a calculation of the hadron spectrum, where typical scales are about 1.5 fm. a lattice spacing of about 0.5 fm would seem optimal [43]. The second important constraint is that the simulation must be carried out in the scaling region close to the fixed point of the theory. In this regime, the ratio of physical quantities becomes weakly dependent on the lattice cutoff and thus extrapolation to zero lattice spacing provides a reliable prediction of the ratio in the continuum. The size of the scaling region is limited by the discretisation errors in the lattice action used. For the Wilson action, the scaling region does not extend out to the optimal hadronic scale of 0.5 fm.

Section 1.2.1 introduced Symanzik improvement, a systematic means of removing the finite lattice spacing errors in the action. The technique has been applied to quenched QCD in previous calculations with limited success. The improvement programme relies on lattice perturbation theory to calculate the coefficients of the extra operators in the action. Until the advent of tadpole improvement, lattice perturbative expansions had alway failed at surprisingly small scales, of the order of 0.1 fm [44]. This meant that the coefficients used to try to extend the scaling region of lattice QCD with Symanzik improvement were unreliable, since the effects of the tadpole graphs had not been included. Alford *et.al* [40, 41] introduced a Symanzik improvement scheme incorporating tadpole improvement in the coefficients and it is this action which is tested.

The plaquette, used in the Wilson action and expanded classically in powers of the lattice spacing, gives at lowest order [45]

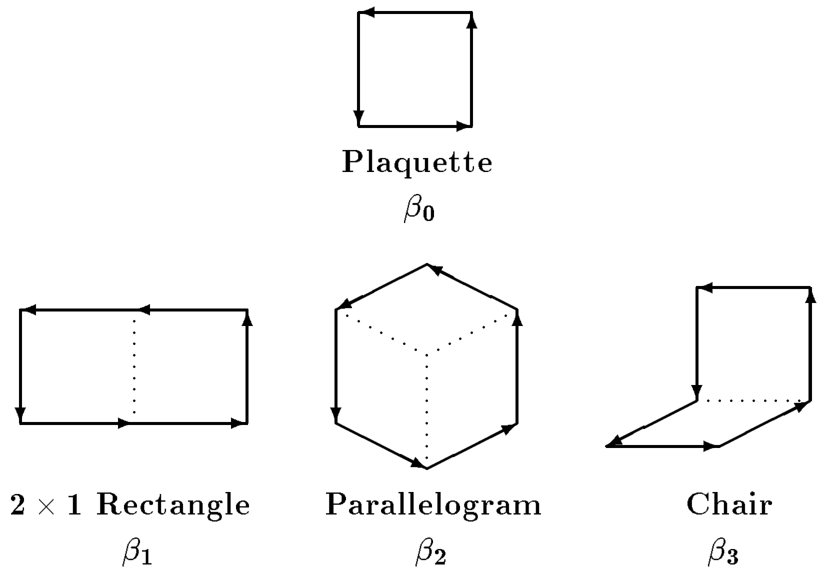
$$\begin{aligned} \sum_{\mu,\nu}\left[1-\frac{1}{3}\mathrm{Re}\ \mathrm{Tr}\ U_{\Box}\right] &= \frac{1}{4}\sum_{\mu,\nu}\mathrm{Tr}\ F_{\mu\nu}F_{\mu\nu}-\frac{1}{24}a^2\sum_{\mu,\nu}\mathrm{Tr}D_\mu F_{\mu\nu}D_\mu F_{\mu\nu} \\ &+ \text{total derivatives}+O(a^4). \end{aligned} \tag{4.1}$$

The second term is the source of the leading discretisation error. From its form, it explicitly breaks continuum Euclidean symmetry and thus this symmetry should be broken on the lattice to $O(a^2)$. Mass ratios of asymptotic states will have errors at the same order.

To remove the $O(a^2)$ errors from the Wilson action built from the plaquette, a dimension 4 operator, a set of dimension 6 operators must be added, with coefficients suitably chosen to remove the artefacts. The dimension 6 operators can be constructed from combinations of the plaquette and the trace of path ordered products around loops of six links. There are three such loops in 4 dimensions; the "2×1 rectangle", the "parallelogram" and the "chair" (see figure 4.1) . Thus the Symanzik-improved action at $O(a^2)$ has, in general, 4 terms.

$$\begin{aligned} S_G = \quad & -\beta_0 \sum \tfrac{1}{3}\mathrm{Re}\ \mathrm{Tr}\ U_{\square} - \beta_1 \sum \tfrac{1}{3}\mathrm{Re}\ \mathrm{Tr}\ U_{\mathrm{rect}} \\ & -\beta_2 \sum \tfrac{1}{3}\mathrm{Re}\ \mathrm{Tr}\ U_{\mathrm{para}} - \beta_3 \sum \tfrac{1}{3}\mathrm{Re}\ \mathrm{Tr}\ U_{\mathrm{chair}}. \end{aligned}$$

Figure 4.1: Wilson Loops used in the Symanzik Improved QCD lattice action.



The pure gauge sector of QCD has one free parameter, the coupling constant, and thus the three extra couplings in the action must be fixed by a perturbative calculation. This analysis is greatly simplified by relaxing the improvement condition to ensure only on-shell quantities are improved. At tree-level, the coefficients are determined by classical improvement of the action. This yields

$$\begin{aligned} \beta_0 = \frac{5}{3}\beta, \quad \beta_1 = -\frac{1}{12}\beta, \\ \beta_2 = \beta_3 = 0. \end{aligned} \tag{4.2}$$

If the action is to $O(a^2)$ improve on-shell quantities only, this choice of coefficients is not unique [45] ; a local infinitesimal change of variable in the lattice path integral can be made which preserves spectral quantities but changes the operator coefficients,

$$\tilde{U}_\mu(x) = e^{\epsilon X_\mu(x)} U_\mu(x). \tag{4.3}$$

X_μ is a traceless anti-hermitian product built from the local links around the plaquette.

Varying ϵ allows the improved action of the form (4.2) to generate an infinite set of improved actions characterised by a free parameter x,

$$\beta_0 = \left(\frac{5}{3} - 24x\right)\beta, \quad \beta_1 = \left(-\frac{1}{12} + x\right)\beta,$$
$$\beta_2 = 0, \quad \beta_3 = x\beta. \tag{4.4}$$

x is only constrained by maintaining a positive definite action which thus has a well defined classical vacuum.

The action now has discretisation errors of the form $O(\alpha_s a^2)$ and $O(a^4)$. The first of these errors arises from operator mixing, leading to a renormalisation of the coefficients away from their classical values at finite coupling, the second comes from higher order discretisation errors. Alford *et al.* suggest that, after tadpole improvement, the $O(\alpha_s a^2)$ and $O(a^4)$ errors are of comparable magnitude. Lüscher and Weisz [45] have calculated the improved action coefficients at one-loop in a theory where two of the four dimensions are compactified with twisted boundary conditions. In such a theory, the gauge bosons acquire mass and become asymptotic states. The coefficients in their calculation are then chosen to remove the $\alpha_s a^2$ errors in the propagators of these states. The field redefinition method is again used to shift the couplings, this time to remove the "chair" term from the action. This choice is a convenient one both for their perturbative calculation and for computer simulations - there are more possible orientations for the "chairs" and hence Monte-Carlo updating with these terms requires more gauge force terms to be computed.

The coefficients in the action are then tadpole improved. At tree level, tadpole improvement simply involves multiplying all the coefficients in the action by $\frac{1}{u_0^N}$

where N is the number of links in the operator. Note that tadpole improvement for the Wilson action can be considered as a renormalisation of the bare lattice coupling, whereas the improved action contains a sum of terms with different numbers of links and thus the couplings are altered relative to one-another. The tadpole coefficient is defined in terms of the plaquette expectation value

$$u_0 = \langle \frac{1}{3}\mathrm{Re\ Tr}\ U_\Box \rangle^{\frac{1}{4}}. \tag{4.5}$$

For higher order perturbative calculations such as the one-loop effects included here, tadpole improvement is more complicated. An n-link operator, S occurring in the action with l-th order perturbative coefficient $c_{(l)}$ is tadpole improved to

$$c_{(l)}(\alpha_s)S[U] \rightarrow c_{(l)}(\alpha_s)\tilde{u}^n_{(l)}(\alpha_s)\frac{S[U]}{u_0^n}, \tag{4.6}$$

with $\tilde{u}_{(l)}(\alpha_s)$ the l-th order perturbative expansion of the mean link tadpole parameter.

The strong coupling constant, $\alpha_\Box$, and the perturbative expansion of the mean link tadpole coefficient are defined from a one-loop perturbative calculation of the plaquette [46]. In perturbation theory, using the Wilson action, the expansion is

$$\langle \frac{1}{3}\mathrm{Re\ Tr}\ U_\Box \rangle = 1 - \frac{4\pi}{3}\alpha_\Box. \tag{4.7}$$

For the tree-level improved action, defined in (4.2) the coefficient in the plaquette has been computed in [46] as

$$\langle \frac{1}{3}\mathrm{Re\ Tr}\ U_\Box \rangle = 1 - \frac{4\epsilon\pi}{3}\alpha_\Box. \tag{4.8}$$

with $\epsilon = 0.732524$. Thus the strong coupling constant, defined in terms of the logarithm of the plaquette, is

$$\alpha_\Box = \frac{-\ln\langle \frac{1}{3}\mathrm{Re\ Tr}\ U_\Box \rangle}{3.06839}. \tag{4.9}$$

The one-loop perturbative expansion of the plaquette then gives

$$u_{(1)} = 1 - 0.767098\alpha_{\Box}. \tag{4.10}$$

The couplings for the one-loop tadpole-improved action are then (expressed in terms of the plaquette coupling β_0 for simplicity)

$$\begin{aligned} 2\times 1 \text{ Rectangle } \beta_1 &= -\frac{\beta_0}{20u_0^2}(1+0.4805\alpha_{\Box}) \\ \text{Parallelogram } \beta_2 &= -\frac{\beta_0}{u_0^2}\ \ 0.03325\alpha_{\Box}. \end{aligned} \tag{4.11}$$

4.1.1 The Transfer matrix for Improved Actions.

As discussed in chapter 1, the transfer matrix formalism links the statistical mechanics system simulated on computers with the quantum theory. The analysis is clearly defined for actions where each timeslice on the lattice is linked to its nearest neighbours only, as is the case with the Wilson action. The Symanzik-improved action with Euclidean symmetry necessarily includes the 2×1 rectangle term which couples timeslices separated by two lattice spacings. The construction discussed in section 1.1.4 breaks down since it relies on a clear definition of a complete set of states of the gauge links on one timeslice, which can no longer be made. Lüscher [47] demonstrated that the definition of the transfer matrix can be extended and correctly defined for the improved theory, which thus has a well defined link with continuum QCD. Unfortunately, the transfer matrix is not now (in general) hermitian for simulations at finite lattice spacings. The new construction does demonstrate that the mass spectrum can be still computed from the exponential decay of two-point correlation functions, but these decays may involve damped oscillatory behaviour. With a hermitian transfer matrix, an effective mass for a ground state, calculated from a two-point function between identical source-sink operators (for example, glueball operators built from plaquette-plaquette correlation functions) could be shown to converge to its infinite time-separation value from above. For a particle state that can be built from a variety of lattice operators, a variational technique can then be employed to approximately diagonalise the transfer matrix and reliably extract masses for small source-sink separations. The monotonic approach from above is no longer necessarily true and hence the reliability of this variational calculation for improved actions is questionable. The technique may still be practical for cases where the coherence length of any oscillatory modes is small.

4.2 The Glueball Spectrum from the Lattice

Glueballs are postulated as bosonic, physical, bound states of the gluon fields [48]. There has to date been no experimental confirmation of their existence. However the search for them may be hampered by the nature of their interaction with hadronic states. The mass of the lightest glueball state, the scalar (0^{++}) glueball, has been predicted from lattice calculations [49, 50] at about 1500 MeV. The existence of data from calculations with the Wilson action at small lattice spacings allows comparative testing of the improvement to be performed.

The tadpole-improved action was tested with a Monte-Carlo calculation of the glueball spectrum of quenched QCD. Glueballs are composed of gluonic fields only (in the quenched approximation) with no fermion component. This gives an independent test of the scaling properties of the scheme without the need to disentangle the discretisation errors inherent in fermion simulations (the Wilson and S-W action have leading errors at lower order than the improved action, $O(a)$ and $O(\alpha_s a^2)$ respectively and these errors remain unaltered by the gauge field improvement scheme discussed [51]). Since glueballs are predicted as physical states of QCD, their masses are on-shell quantities and as such, should be improved to the same order as the action.

Glueball spectroscopy on the lattice has proven technically difficult in the past. The source of this difficulty is the reliable extraction of Monte-Carlo signals for asymptotic glueball states. The vacuum fluctuations of the massless gluon fields are large and yet masses predicted for the glueball states are high. Thus the Monte-Carlo signal is rapidly lost for all but the smallest lattice spacings. In previous lattice calculations, the signal was improved by creating glueball states from "fuzzy" link variables [52]. Here, the bare lattice links are enhanced with some amount of the sum of staples around the link. Glueball operators are then made up from the sum of traces of path-ordered products around loops constructed from the new link variables. These "fuzzed" operators then have a higher overlap with the glueball states, and so correlation functions should approach their asymptotic decay before the signal is swamped by the vacuum fluctuations. This approach can be combined with a multistate variational calculation.

For this calculation, no fuzzing of the gauge links was performed. On the coarse lattices used in these simulations, the glueballs should be only a few lattice spacings across, and as such should have a good overlap with operators built from small Wilson loops.

4.2.1 Lattice Glueball States

In discretising Euclidean spacetime, the continuous symmetry group $O(4)$ is broken. All that remains of it is the discrete group, O_4. In the continuum, glueball states transform as irreducible representations (irreps) of the continuous $O(3)$ group of spatial rotations and thus have a spin quantum number, J according to the irrep concerned. States also have two quantum numbers, with value ± 1 according to how they transform under parity and charge conjugation operations. Thus for each spin combination, there are 4 PC (parity, charge conjugation) irreps written J^{PC}.

On the lattice away from the continuum limit, glueballs transform according to the irreps of the reduced symmetry point group, O. A more detailed explanation of point groups can be found in *eg.* [53], their application to glueball spectroscopy is discussed at length in [42] and summarised here. The group, O has 24 elements which can be classified into 5 irreps, labelled A_1, A_2, E, T_1, T_2 of dimension $1, 1, 2, 3, 3$ respectively. Parity (a spatial symmetry) can be added naturally by extending the group to O_h to include reflections. The 10 irreps of this extended group are generated simply from the irreps of O. In the continuum limit, the representations formed by O_h do not specify the spin; in general, a lattice irrep will contain a range of possible spin states. There are however a set of selection rules governing which spin occurs in which lattice irrep. The lattice state belonging to lattice irrep R, $|\psi\rangle_R$ has contributions from continuum spin states, $|\psi\rangle_{J,m}$.

$$|\psi\rangle_R = \sum_{J,m} c^R_{J,m} \; |\psi\rangle_{J,m}. \tag{4.12}$$

The selection rule imposes the constraint that spin J will only appear if

$$D_J^0 \supset R, \tag{4.13}$$

where D_J^0 is the subduced representation of irrep D_J. These selection rules are simple for spins 0,1 and 2. In particular, in the continuum limit,

$$\begin{aligned} m(A_1) \quad &= \quad M(0), \\ m(E) = \quad m(T_2) \quad &= M(2), \end{aligned} \tag{4.14}$$

where $m(R)$ are lattice ground state masses and $M(J)$ the lightest continuum glueball masses.

This implies:

- The lightest state of the A_1^{++} operator on the lattice should correspond to the continuum 0^{++} (scalar) glueball (predicted as the lightest).
- The lightest states of the E^{++} and T_2^{++} should be degenerate if the continuum Euclidean symmetry is restored.

This gives a test of two proposed properties of the improved action - a scaling window out to ≈ 0.5 fm giving reliable predictions of the scalar glueball mass and restored Euclidean symmetry giving degenerate E and T_2 lattice glueball states. The Monte-Carlo calculation described in this chapter attempted to address both of these points, however no reliable glueball signal for the tensor (E^{++} and T_2^{++}) states was found. The data for the tensor glueball states are not presented.

A set of n glueball creation operators $\{O_i(t)\}$ is built from combinations of Wilson loops on timeslice t which transform under the action of the discrete lattice symmetry according to the irrep of the glueball state. This set is used to form an incomplete basis for the quantum state and then a variational technique is employed to attempt to form an approximate ground state. For large lattice spacings, a glueball should be only a few lattice points in size (in accordance with the discretisation scheme argument presented earlier). In the glueball calculation on the coarse lattice, operators built from unfuzzed loops of up to 4 lattice points across are used (see figure 4.2). This choice was made to give access to a range of sizes up to a few grid points across to ensure at least one operator with a good groundstate overlap. The particular shapes (squares, rectangles, chairs and parallelograms) were chosen since the group theory calculations to generate glueball operators from these shapes is published elsewhere [42].

The QCD action and thus the vacuum of the theory transforms like an A_1^+ irrep of the O_h group and has positive charge conjugation quantum number. The A_1^{++} glueball operators will have an overlap with the QCD vacuum and thus in general they have a non-zero expectation value. The lightest glueball state, the 0^{++} is then the mass gap above the vacuum. To convert the A_1^{++} operators into excited state creation operators, their vacuum expectation values must be subtracted off. Thus these operators are defined as

$$O_{A_1^{++}} = \tilde{O}_{A_1^{++}} - \langle \tilde{O}_{A_1^{++}} \rangle \tag{4.15}$$

with $\tilde{O}$ the real part of the trace of the appropriate Wilson loop averaged over all lattice orientations.

For any state, an $n \times n$ correlator matrix is then built from the set of lattice glueball operators with appropriate quantum numbers and lattice symmetry.

$$C_{ij}(\Delta t) = \sum_t \langle O_i(t)\, O_j(t+\Delta t) \rangle. \tag{4.16}$$

The operators do not form an orthonormal basis. To calculate an effective optimal glueball correlator, a constrained diagonalisation (which incorporates the non-orthonormal nature of the creation/annihilation operators) of the correlator on timeslice $t_{\rm diag}$ is performed to give an optimised combination of states, $\mathbf{v}$. The ground state will have the highest large-time correlation and so the optimised vector should be chosen to maximise λ in

$$C(0)\mathbf{v} = \lambda\, C(t_{\rm diag})\, \mathbf{v}. \tag{4.17}$$

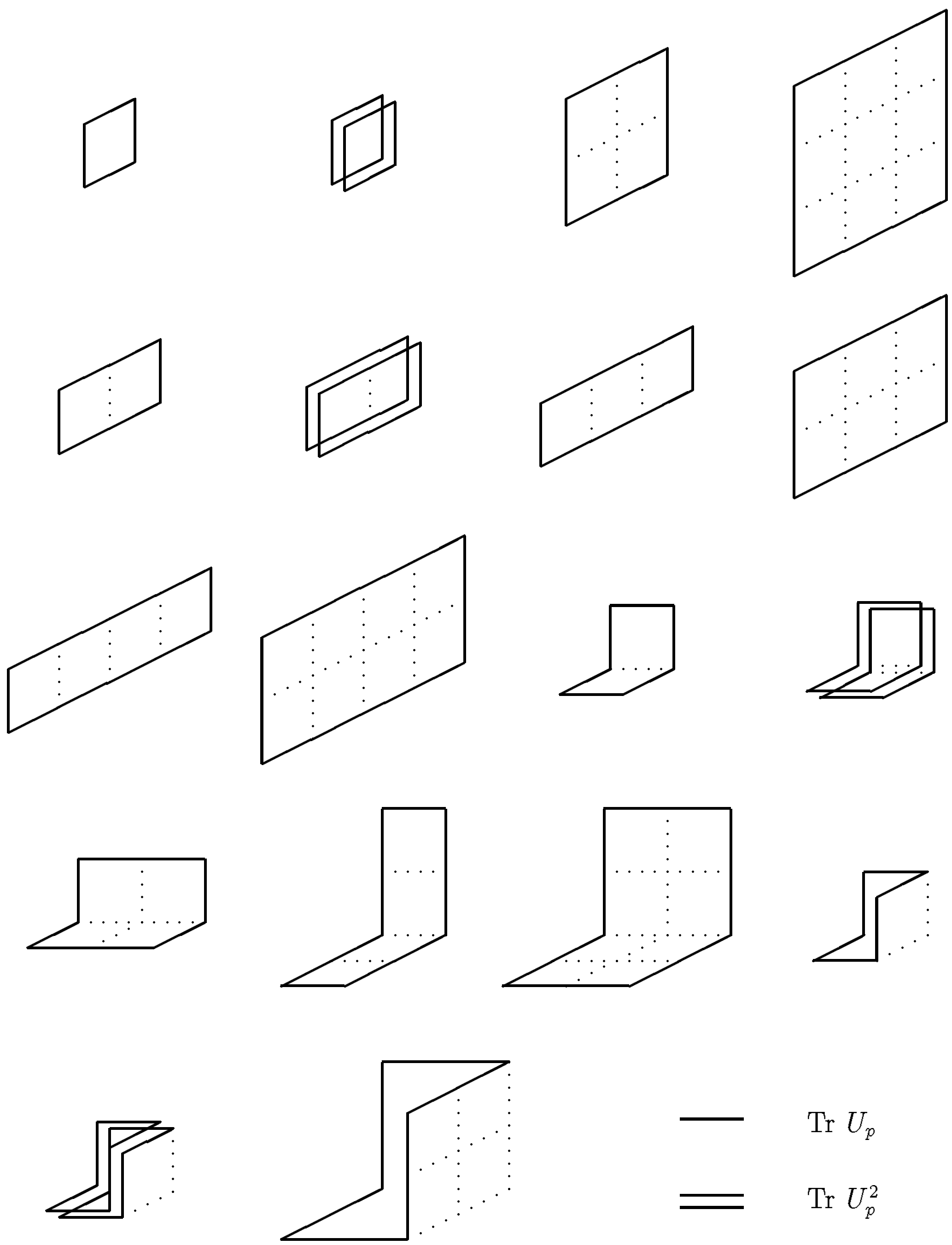


Figure 4.2: Wilson loops used in glueball operators

The vector, $\mathbf{v}$ is normalised such that

$$\mathbf{v}^T\, C(0)\, \mathbf{v} = 1. \tag{4.18}$$

A one-dimensional glueball approximate ground-state correlator is then computed and its asymptotic decay investigated.

$$G(t) = \mathbf{v}^T\, C(t)\, \mathbf{v}. \tag{4.19}$$

Note that the non-hermitian nature of the transfer matrix, as discussed in section 4.1.1 means that the effective mass extracted from the correlator G does not necessarily approach its asymptotic value monotonically from above.

4.3 Simulation Details

The Monte-Carlo calculation was performed on DEC Alpha and Hewlett-Packard workstations as well as using 3000 CPU hours of Cray-T3D time. The T3D was used as a "task-farm" so there were no off-node communications. The philosophy of the calculation was to check the reliability of QCD calculations based on lattices accessible to workstations (*ie.* large lattice spacing and thus a small number of lattice points) and the T3D was employed essentially as a set of 64 isolated workstations.

Two runs, at different lattice spacings were performed using the improved action at plaquette couplings of $\beta_0 = 6.8$ and 7.4. One simulation with the Wilson action at $\beta_W = 5.5$ was performed to complement the existing Wilson glueball data and estimate the scaling violations for large lattice spacings for that action. The Wilson run was performed on a lattice with spacing (calculated from the static quark potential) similar to the $\beta_0 = 7.4$ lattice.

The gauge fields were updated with both Cabibbo-Marinari [CM] (pseudoheat-bath) and over-relaxation [OR] techniques. Timings for the different update methods on a DEC alpha workstation are given in table 4.1. Note that updates with the improved action are five times more costly than Wilson action updates in the workstation code. This extra overhead arises from the need to calculate the force term on a link which now involves the multiplication of 156 3×3 complex matrices whereas for the Wilson action, only 12 such multiplications are required. Since the code was written for serial processing, the sites of the lattice are looped over sequentially. This incurs a large overhead for the improved action since the number of subroutine calls in the sitewise implementation is orders of magnitude higher than a parallel code. The serial code could thus be optimised by unwrapping some of the low-level update subroutines. The improved action can not be implemented as efficiently as the Wilson code on parallel computers since it can not be red-black preconditioned. In Wilson simulations, the gauge fields are divided into sub-lattices, each consisting of all the links in a given direction, μ from sites of one lattice parity index, p ('parity' here implies the lattice is divided into two even-odd site checkerboards). All the links on a sub-lattice can then be updated simultaneously whilst the other fields are held as a constant background. For the new action with six-link terms, fields on a sub-lattice have direct interactions and can not be updated in parallel. A more coarse preconditioning could, however, be implemented. Since the main overhead in the glueball calculation is the gauge field update process, glueball correlators were measured frequently. One CM sweep and one OR sweep were performed between measurements. As a result, each sample does not represent a statistically independent data point.

Table 4.1: CPU times on DEC Alpha-Workstation (`meursault@ph.ed.ac.uk`)

Action	Time per site (ms/site)
Cabibbo Marinari update (Improved action)	3.6
Over-relaxation update (Improved action)	3.3
Cabibbo Marinari update (Wilson action)	0.8
Over-relaxation update (Wilson action)	0.4
Calculate Glueball correlators (along one T-axis)	1.4

As these measurement are highly correlated, the data were added into bins of 1000. To ensure no observable autocorrelations remained, the errors in all quantities were calculated using both a one- and two-point jack-knife method over the bins and checked for consistency. No residual autocorrelation effects were found. The signal was enhanced by exploiting the lattice Euclidean symmetry of the action. This allows any of the four space-time axes to be regarded as the time axis and thus correlations were measured for source/sink operators separated along all four axes. These measurements will not provide independent statistical data but the use of large data bins removed the correlations found in the four channels.

The value of the tadpole coefficient, u_0 required as an input parameter to the action is not known *a priori*. A self-consistent solution, where the input mean link value used in the action matches the fourth root of the plaquette measured with that action must be sought. Approximate values for u_0 are found in [41]. These were taken as starting guesses and the plaquette calculated for these inputs to 4dp. A range of input values close to the estimated self-consistent point were then used, leading to the fixed-point results given in table 4.2. Solutions, correct to 3dp. were thus obtained. The glueball program calculates the plaquette as a by-product and the glueball runs then gave the plaquette values in the table accurate to 5dp. Morningstar [54] suggested a more systematic approach to this tuning problem using the approach of Ferrenberg and Swendsen [55] to compute the plaquette simultaneously for a set of theories close to the initial estimated self-consistent point. At any rate, the tuning process require small amounts of computer time.

Table 4.2: Simulation parameters for the improved action.

Lattice	β_0	β_1	β_2	u_0^4	$\langle$Plaquette$\rangle$
6^4	6.8	-0.5568	-0.0821	0.467	0.46692(1)
8^4	7.4	-0.5424	-0.0634	0.555	0.55516(1)

4.4 Setting the Scale; String Tension Measurements

The lattice spacing for these simulations was fixed from consideration of the static quark potential. For a confining theory such as QCD, the potential at large inter-quark separation becomes linear and the constant of proportionality is called the string tension. The string tension is determined from potential model fits to experimental data and is determined as $\sqrt{\sigma} = 440$ MeV. Measuring this quantity on the lattice allows a non-perturbative determination of the scale.

On the lattice, the static quark propagators are path-ordered products of timelike link variables connecting the source to the sink (at the same spatial position). The spin structure of the quarks decouples at infinite quark mass and thus the quark anti-quark pair acts simply as a colour dipole source. The energy of the system is then just the inter-quark potential. The pair can be adiabatically created at separation $\mathbf{r}$ by linking the two static fermion propagators with a path-ordered product "string" of gauge links.

The string tension is determined by fitting the data for a quark anti-quark pair separated along an axis of the lattice to the large separation predicted function

$$V(r) = \sigma r - \frac{\pi}{12r} + b. \tag{4.20}$$

The linear coefficient (measured in lattice units) then corresponds to σa^2. Thus the lattice spacing can be determined by fixing $\sqrt{\sigma}$ to its "physical" value, 440 MeV.

Figures 4.3 and 4.5 show the effective mass of the $Q\bar{Q}$ pair on the two improved action lattices. Figure 4.7 shows the data generated with the Wilson action at $\beta_W = 5.5$. Wilson loop data were combined into bins of 100 measurements, with each measurement separated by 1 OR and 1 CM lattice sweep. For the $\beta_0 = 6.8$ and 7.4 runs, 60 and 30 bins were generated respectively. The Wilson $\beta_W = 5.5$ run was performed on 50 bins. All measurements were checked for autocorrelations by increasing the bin size.

For the $\beta_0 = 6.8$ data, good estimates of the static quark pair effective mass out to timeslice 3 are obtained for quarks of separation up to $(2, 1, 0)$. The $(3, 0, 0)$ system has extreme finite volume effects at this separation. A good plateau is seen in the data at timeslice 2 and this is used in the determination of the string tension. Figure 4.4 show this data along with the fit to (4.20).

The $\beta_0 = 7.4$ data gives a good plateau on timeslice 3. The $(4,0,0)$ point has significant finite volume effects on timeslice 4 and is not included on the plot. Figure 4.6 shows the data from timeslice 3 and the string tension fit.

The $\beta_0 = 5.5$ Wilson data are similar to the improved $\beta_0 = 7.4$ data as anticipated, since the two were chosen to have similar lattice spacings. The fit shown on figures 4.6 and 4.8 is to the ansatz given in (4.20) for on-axis data points only. These two couplings correspond to roughly equivalent lattice spacings (≈ 0.27 fm.). The off-axis points for the improved action lie closer to the fit line than their Wilson action counterparts, indicating a reduction in the discretisation errors that lead to a breaking of Euclidean symmetry.

Results for the scale from the different simulations are given in table 4.3. The statistical errors in the data are small. The errors quoted for the scale come from an assessment of the systematic errors in the data. This assessment attempted to incorporate effects from symmetry breaking, problems with the ansatz (*eg.* the coulombic term colour charge, $\pi/12$ is an infinite separation limit) and residual excited state contamination in the effective mass plots. String tensions were calculated by adding in these effects and noting the change in the measured lattice value of σa^2.

	$\beta_W = 5.5$ (Wilson)	$\beta_0 = 6.8$ (Improved)	$\beta_0 = 7.4$ (Improved)
σa^2	0.37 ± 0.01	0.91 ± 0.02	0.36 ± 0.01
a (fm.)	0.272 ± 0.004	0.427 ± 0.005	0.269 ± 0.004

Table 4.3: Scales from string tension measurements

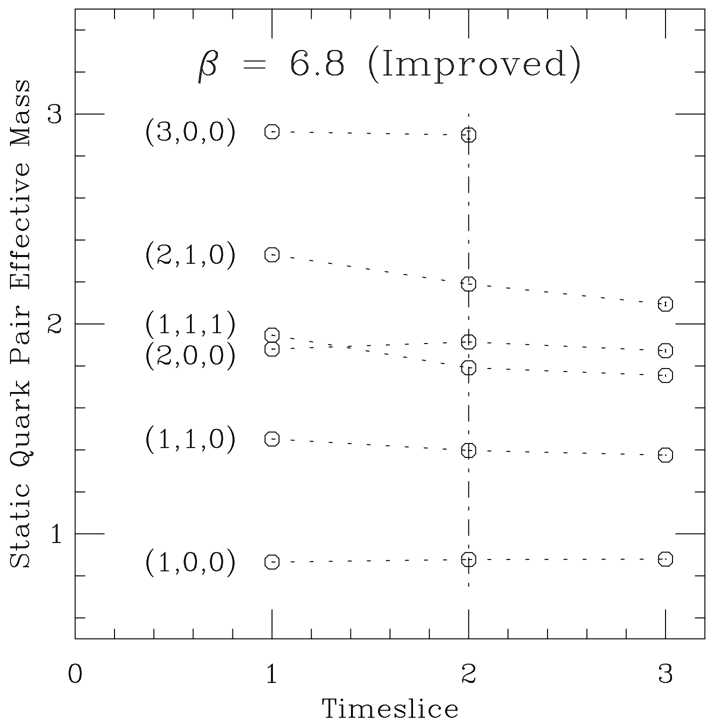


Figure 4.3: Effective mass of a $Q\bar{Q}$ pair, Improved action $\beta_0 = 6.8$

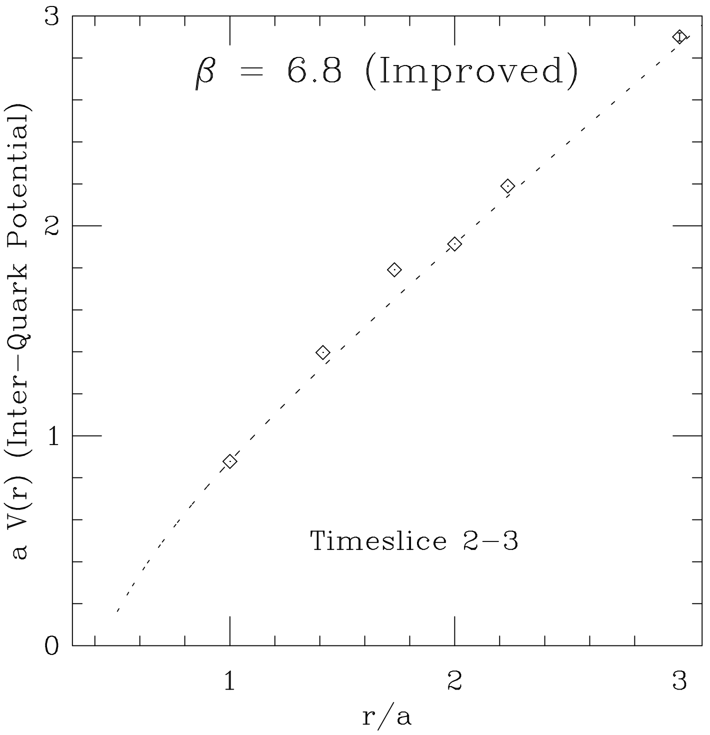


Figure 4.4: Static Quark Potential, Improved action $\beta_0 = 6.8$

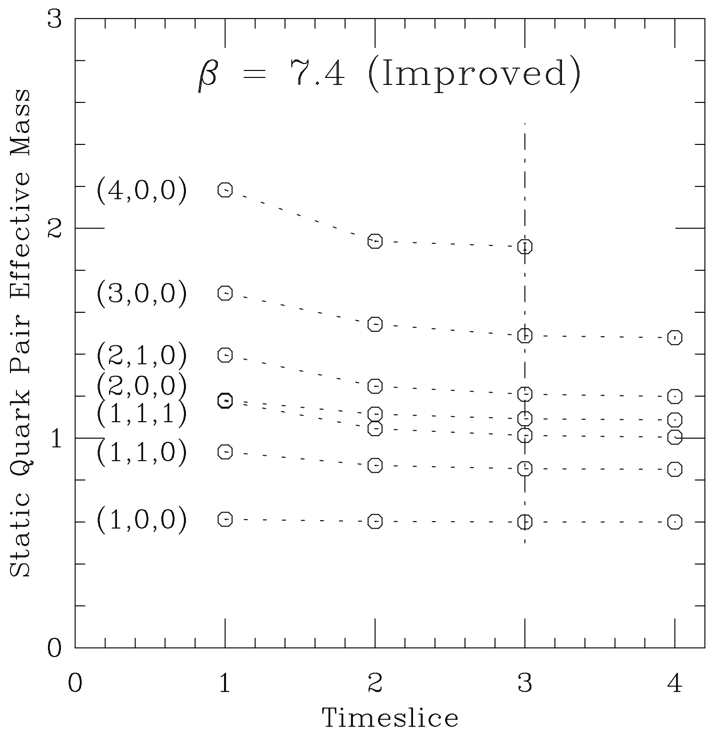


Figure 4.5: Effective mass of a $Q\bar{Q}$ pair, Improved action $\beta_0 = 7.4$

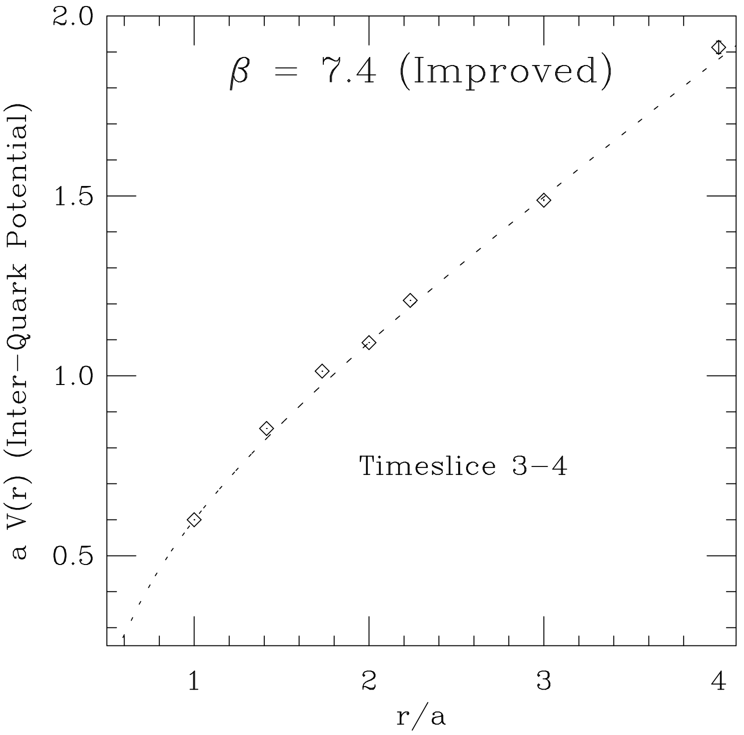


Figure 4.6: Static Quark Potential, Improved action $\beta_0 = 7.4$

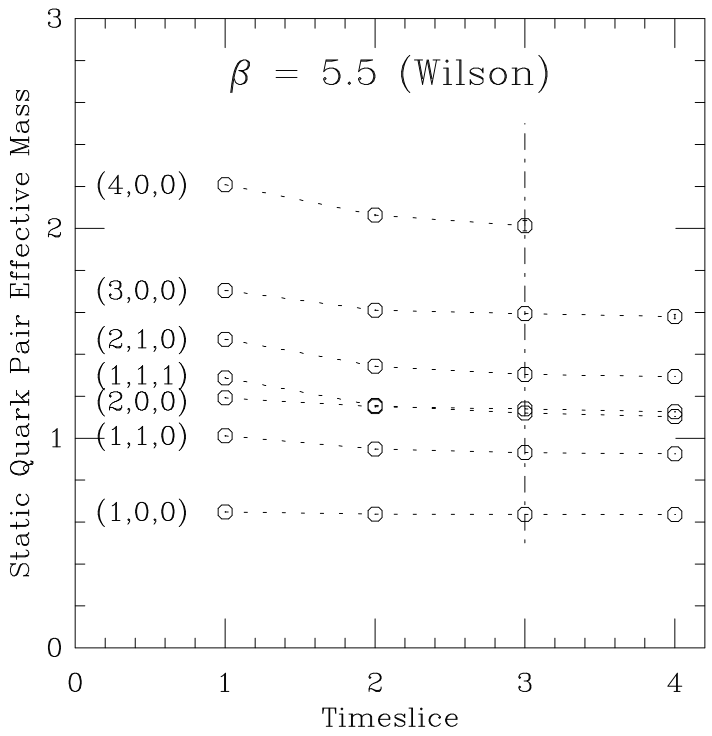


Figure 4.7: Effective mass of a $Q\bar{Q}$ pair, Wilson action $\beta_W = 5.5$

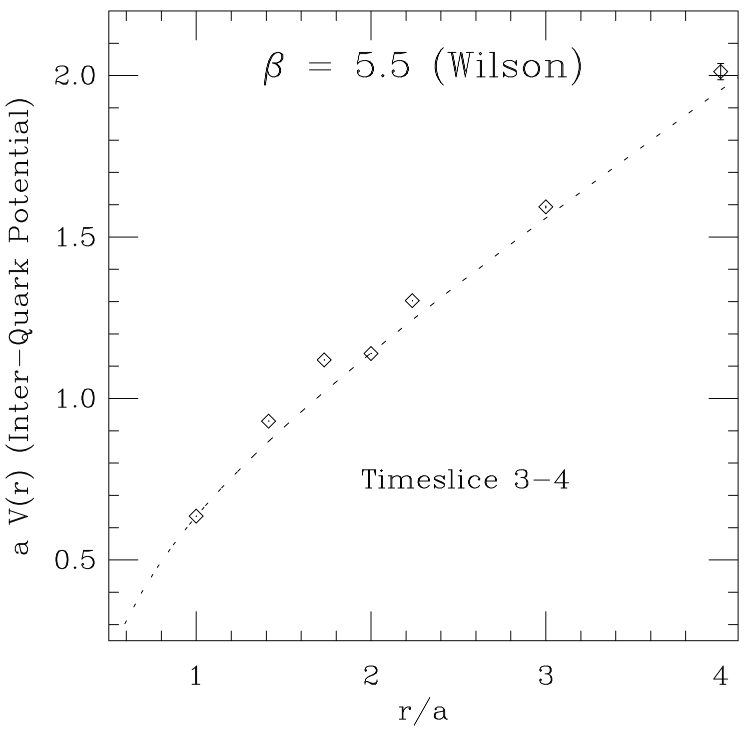


Figure 4.8: Static Quark Potential, Wilson action $\beta_W = 5.5$

4.5 Glueball Results

The scalar glueball mass was calculated on a different Markov chain from the static quark potential to simplify the error analysis.

4.5.1 A_1^{++} with the Improved Action

Figures 4.9 and 4.10 show effective mass plots for the scalar (A_1^{++}) lattice glueball. Circles indicate data from the decay of a correlator optimised according to (4.17) on timeslice 1 and crosses indicate data optimised on timeslice 2.

For the smaller of the two lattice spacings ($\beta_0 = 7.4$) the effective masses on timeslice 1, 2 and 3 are consistent with a plateau. Note that the approach to the plateau is from below. Changing the diagonalising timeslice does not significantly alter the resulting effective mass plot, giving confidence that the variational procedure has produced a good approximation to the ground state wavefunction on timeslice 1, in spite of the theoretical problems with defining the technique rigorously. When the static potential scale is introduced, the length of the plateau in physical units is about 0.7 fm, longer than plateaux in other glueball calculations.

At the larger lattice spacing, the conclusions to be drawn from the data are less clear. Altering the diagonalising timeslice causes a more noticeable shift in the data; however, the two diagonalisation channels are consistent within errors. Again, this implies the variational procedure has helped to give a better ground state wavefunction. It is rather more difficult to make claims about the existence of a plateau since the data on timeslice 2 is so noisy.

4.5.2 A_1^{++} with the Wilson Action

One run of the glueball code was performed with the Wilson action at coupling $\beta_W = 5.5$ both to check against existing glueball data and to complement this data to be used as a comparison with improved action results. The A_1^{++} glueball effective mass for the optimised correlator is shown in figure 4.11. The data on timeslices 2 and 3 seems consistent with a plateau. The coupling was chosen such that the lattice spacing from the static quark potential was close to the lattice spacing from the $\beta_0 = 7.4$ improved run and yet the glueball is significantly lighter for the Wilson action. Notice also the approach to the plateau from above.

4.5.3 A_1^{++} masses

The results from the three runs are summarised in table 4.4. All masses are in lattice units. The data in bold type are those used to produce the final mass estimates (given in the last line of the table). The errors in the physical mass estimates include the effects of both statistical errors in the glueball calculation and scale errors from the string tension measurements. Figure 4.12 shows the

Table 4.4: Scalar glueball masses from diagonalisations on timeslices 1 and 2.

Timeslice	$\beta_W = 5.5$	$\beta_0 = 7.4$	$\beta_0 = 6.8$
$T_{\text{diag}} = 1$			
1-2	1.23 ± 0.02	1.80 ± 0.025	$\mathbf{3.00 \pm 0.11}$
2-3	1.15 ± 0.04	$\mathbf{1.78 \pm 0.14}$	2.6 ± 0.8
3-4	$\mathbf{1.10 \pm 0.09}$	1.63 ± 0.42	N/A
$T_{\text{diag}} = 2$			
1-2	1.23 ± 0.02	1.78 ± 0.024	2.86 ± 0.10
2-3	1.15 ± 0.04	1.78 ± 0.13	2.3 ± 0.6
3-4	1.10 ± 0.09	1.64 ± 0.41	N/A
A_1^{++} mass (lattice)	1.10 ± 0.09	1.78 ± 0.14	3.0 ± 0.1
A_1^{++} mass (MeV)	800 ± 70	1310 ± 100	1380 ± 50

data with existing lattice results from a variety of sources. The $O(a^2)$ scaling violations are clearly demonstrated for the Wilson action. The dashed line is from reference [49] and is a fit to the expected finite lattice mass generated by the Wilson action with $O(a^2)$ discretisation errors. The $\beta_0 = 6.8$ and 7.4 improved action data points are consisent within errors in spite of the large lattice spacings used for both calculations. They are, however, significantly below the $a \to 0$ Wilson action extrapolation. This may be due to inconsistencies in defining the scale from the string tension on the coarse lattice not uncovered in the systematic error analysis.

4.5.4 Tensor glueball states

An attempt to extract signals for 2^{++} (tensor) glueballs was made. No reliable signal was seen in these channels for the large lattice spacings used. Possible improvements to the glueball program which may allow ground state masses to be extracted from the lattice E^{++} and T_2^{++} operators are discussed in section 4.6.

With these masses, a check on the restoration of Euclidean symmetry at finite lattice spacing can be made.

4.5.5 Finite Volume Effects

The improved action calculations were performed on lattices with large physical volumes and thus finite volume effects were anticipated to be small. For the $\beta_0 = 6.8$ data, the lattice extent (in physical units) is 2.6 fm, while for the $\beta_0 = 7.4$ data, it is 2.2 fm. These sizes are comparable to the Wilson action runs performed at small lattice spacings by refs. [49, 50]

Lüscher [56] estimated the effects on the scalar glueball mass extracted from a finite volume calculation. Defining $z = m_0 L$, the mass on a finite lattice is

$$\frac{m(z)}{m(\infty)} = 1 - \frac{g e^{-\sqrt{3}z/2}}{z}. \tag{4.21}$$

and using the estimate of the scalar glueball coupling from [50], $g = 700 \pm 400$ The finite volume effects for the improved action runs (where z for the $\beta_0 = 7.4$ run is 14.4 and for the $\beta_0 = 6.8$ run, $z = 18$) are negligible compared to statistical errors.

With the Wilson action run at $\beta_W = 5.5$, the light glueball induces a larger finite volume effect. The glueball mass may be underestimated due to finite volume effects by as much as 4%. The mass should then be corrected upwards by ≈ 40 MeV. This correction is less than the errors from the calculated estimate and does not account for the large systematic error on the result at this large lattice spacing.

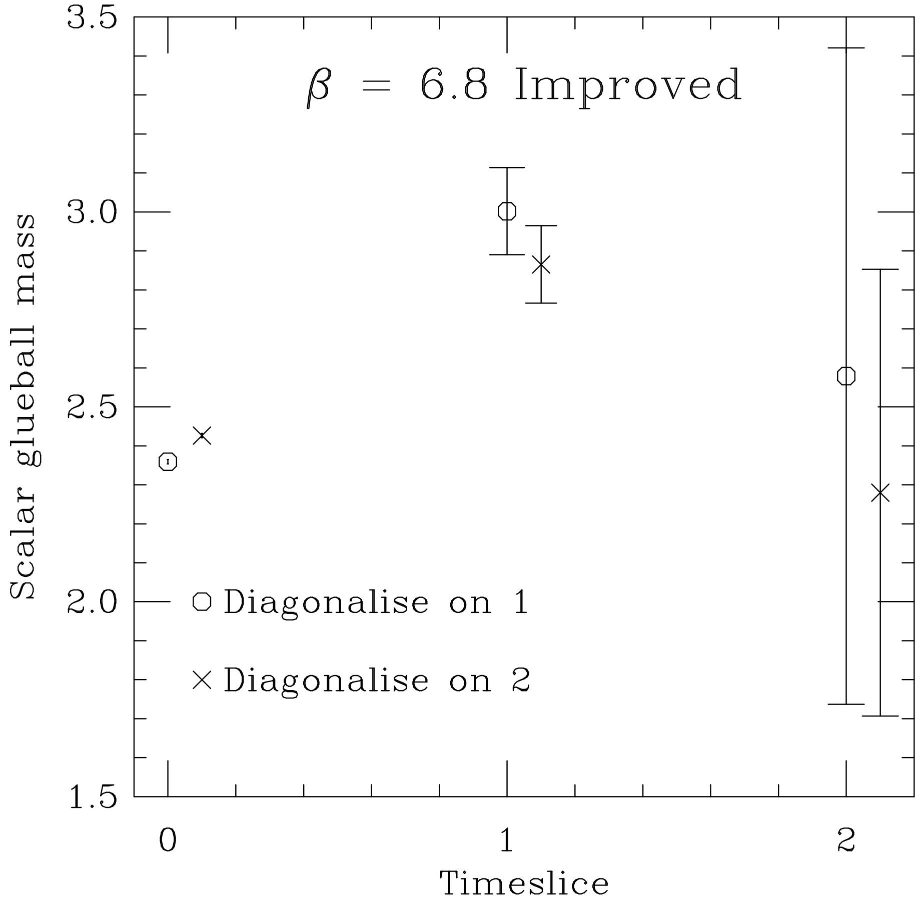


Figure 4.9: A_1^{++} (Scalar) Glueball Effective Mass Plot, Improved action $\beta_0 = 6.8$. Data are from 334 bins of 1000 measurements on a 6^4 lattice. Measurements are separated by one CM and one OR update sweep. Symbols ($\circ, \times$) indicate the diagonalisation timeslice as defined in (4.17)

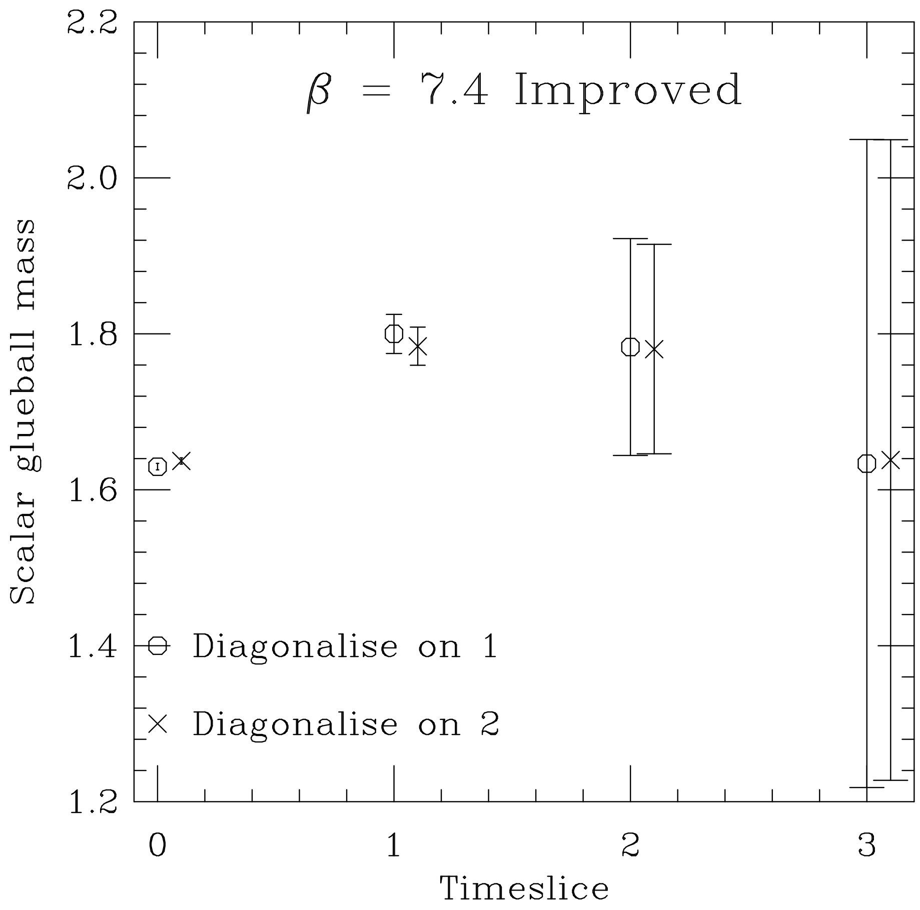


Figure 4.10: A_1^{++} (Scalar) Glueball Effective Mass Plot, Improved action $\beta_0 = 7.4$. Data are from 174 bins of 1000 measurements on a 8^4 lattice. Measurements are separated by one CM and one OR update sweep. Symbols ($\circ, \times$) indicate the diagonalisation timeslice as defined in (4.17)

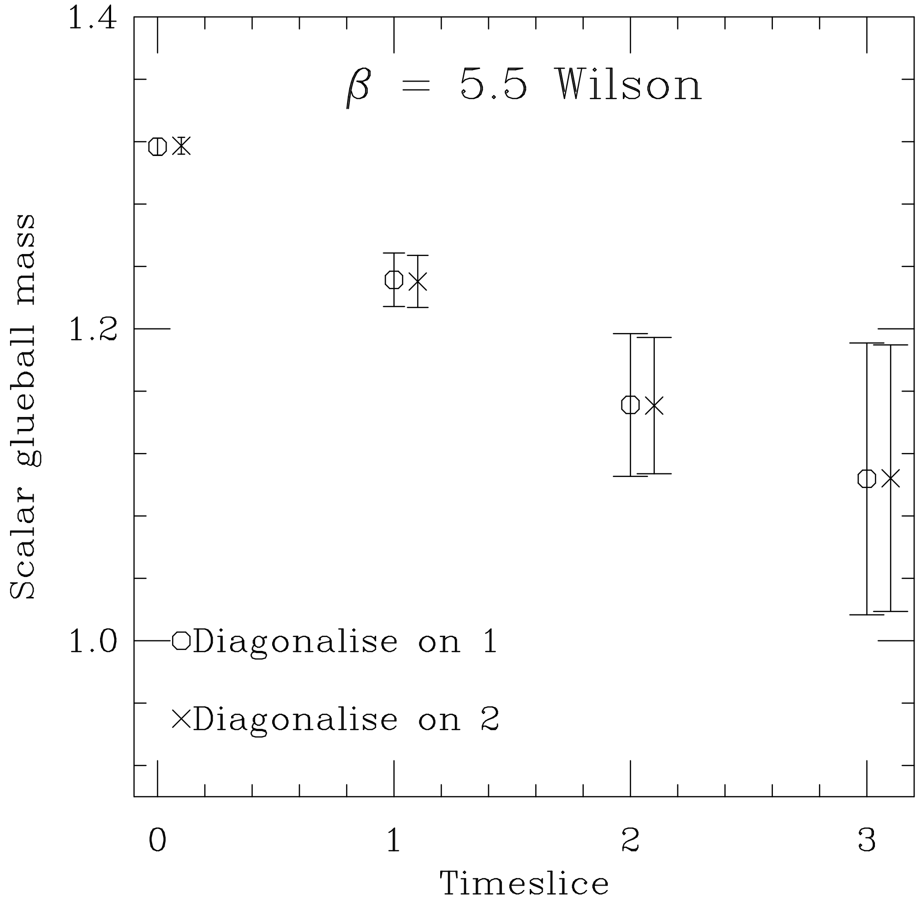


Figure 4.11: A_1^{++} (Scalar) Glueball Effective Mass Plot, Wilson action $\beta_W = 5.5$. Data are from 104 bins of 1000 measurements on a 8^4 lattice. Measurements are separated by one CM and one OR update sweep. Symbols $(\circ, \times)$ indicate the diagonalisation timeslice as defined in (4.17)

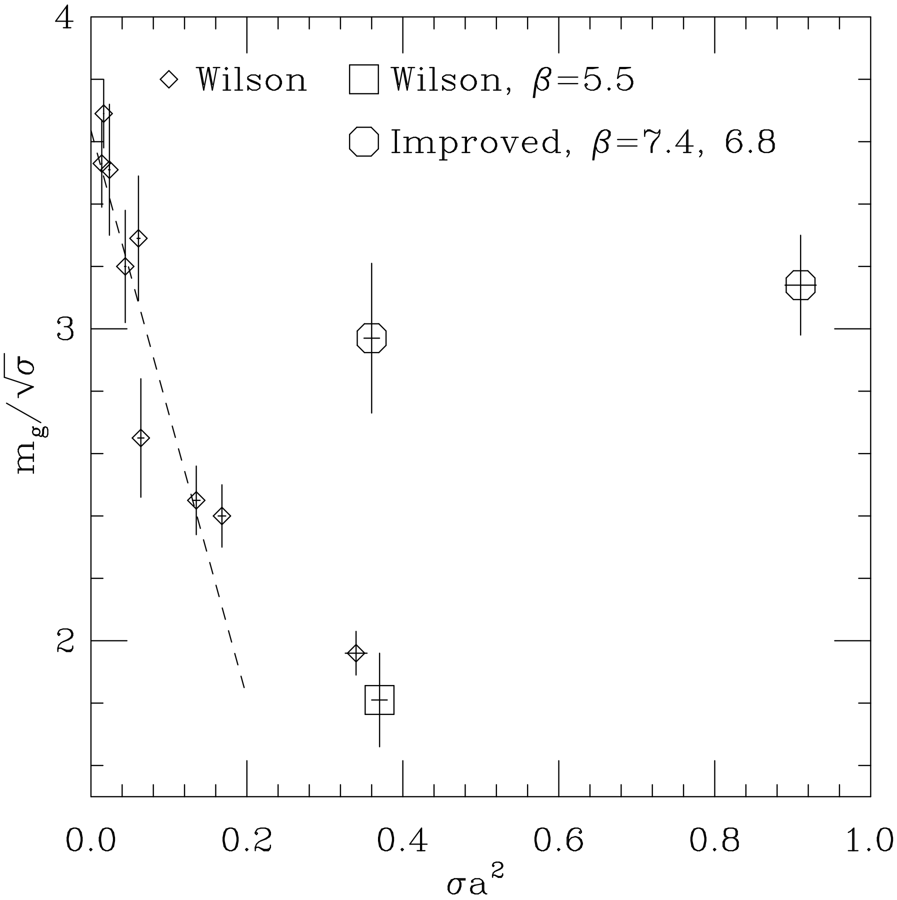


Figure 4.12: Scaling Plot for the Scalar Glueball - Wilson and Improved action.

The Wilson data (◇) are from C. Michael and M. Teper [57, 58], P. de Forcrand *et al.* [59, 60], UKQCD Collaboration [49], Chen *et al.* [50] and this work (□). Improved data are indicated by circles (○).

4.6 Conclusions

The test of the improved action has demonstrated significantly improved scaling behaviour over the Wilson action at a lattice spacing of 0.24 fm. The glueball masses from the two improved simulations performed are consistent within errors. They both lie about two standard deviations from the continuum Wilson limit. This may be due to unforeseen systematic errors in the string tension scale estimates (the techniques for extracting the string tension for small lattice spacing Wilson calculations are significantly more sophisticated) or from pathological behaviour in the transfer matrix. The plateaux for the two calculations (as, it appears, with most Wilson calculations) do not extend far enough to be entirely convincing.

The extra computational cost of simulating with the new action is a five-fold increase in site-by-site update CPU time. The improved action uses far fewer sites to give a reliable mass prediction, however, and a dramatic gain in CPU time in calculations of QCD masses is anticipated. The current implementation of the configuration generation code is not optimised for the simulation of the additional terms in the action and it may be the case that a significant improvement in performance can be made. The optimisation should examine the number of low-level FORTRAN subroutine calls required. A study of the autocorrelations in update algorithms at large lattice spacings would also help to reveal the optimal choice of over-relaxation and Cabibbo-Marinari updates to perform between measurements. In spite of this increased overhead, the improved action has made reliable glueball calculations accessible to contemporary workstations.

The action has been shown to have some significant shortcomings for glueball calculations. The large lattice spacing means only the lightest (0^{++}) glueball correlators can be studied and then for too few timeslices to observe a good plateau. The variational calculation becomes increasingly unstable at large lattice spacings. The presence of complex eigenvalues in the transfer matrix could be responsible. Morningstar [61] suggested a possible technique for circumventing the problem involving an action that explicitly breaks the lattice symmetry and makes the spatial and temporal lattice spacings, a_s and a_t, different. A coarse spatial lattice (with its computational accessibilty) combined with a finer temporal lattice (to allow a more detailed study of an effective-mass plateau) seems optimal. At tree-level

$O(a^2)$ improvement, this allows an action linking only adjacent timeslices to be employed thus removing the transfer matrix problem. Preliminary results indicate tadpole improvement is again crucial to allow reliable perturbative calculations of the action coefficients.

A more stable tensor signal could be investigated with such an action, since the signal will not be lost in the Monte-Carlo noise after one timeslice on a finer time-discretised lattice. A plateau in the less noisy small-separation region could be induced by re-examining the choice of Wilson loops used in the creation operator set used in the (now properly defined) variational calculation. The dataset we have generated should shed some light on the important Wilson loop shapes and sizes to be included.

Chapter 5

Conclusions

Dynamical fermion simulations are currently being actively studied within the lattice community, as the quenched approximation remains the least well understood of the lattice systematic errors. The increase in computing power has now made preliminary calculations of the effects of quenching on QCD predictions attainable, and many large collaborations (*eg.* [62, 63]) are engaged in large-scale studies. Most studies are still with rather too heavy dynamical fermions [62]

Lüscher's method remains as a viable rival to existing fermion algorithms and progress is slow in assessing how best to implement the method and its many extensions. The algorithm is in its infancy with many proposed ideas remaining as yet, unproven.

As with the HMC algorithm, a set of guidelines for implementing the method will need to be developed for optimisation. The Schwinger model has been shown to be a useful testbed for new ideas and implementations. Some of the results illustrated in chapters 2 and 3 have also been demonstrated for more computationally intensive theories using supercomputing resources. All the fermion algorithm performance results of this thesis were computed on workstations.

This ethos of workstation lattice physics was also employed in testing the tadpole improved action for QCD. Problems with reliable calculations were illustrated within the glueball calculation contained in this thesis but ways around these may exist with the asymmetric improvement scheme proposed by Morningstar.

If the two studies within this thesis can be successfully combined, involving dynamical fermion simulations on coarse lattices, then light dynamical fermion simulations may become accessible, since the coarse discretistion makes the lattice correlation length of the light fermions smaller.

Such a scheme would require tadpole-improved fermion actions to higher order and, if the workstation philosophy is to be preserved, then it seems natural to look for the most efficient algorithms for fermion simulations.

References

[1] H. J. Rothe, *Lattice Gauge Theories* (World Scientific, Singapore, 1992).

[2] I. Montvay and G. Münster, *Quantum Fields on a Lattice* (Cambridge University Press, Cambridge, 1994).

[3] K. G. Wilson, Phys. Rev. D **10**, 2445 (1974).

[4] H. B. Nielsen and M. Ninomiya, Nucl. Phys. B **185**, 20 (1981).

[5] K. G. Wilson, in *New Phenomena in Subnuclear Physics*, edited by A. Zichichi (Plenum, New York, 1975).

[6] B. Sheikholeslami *et al.*, Nucl. Phys. B **259**, 572 (1985).

[7] J. Kogut and L. Susskind, Phys. Rev. D **11**, 395 (1975).

[8] A. Clarke and R. Disney, *Probability and Random Processes for Engineers and Scientists* (Wiley, New York, 1970).

[9] M. Creutz, Phys. Rev. D **21**, 2308 (1980).

[10] N. Cabibbo and E. Marinari, Phys. Lett. B **199**, 387 (1982).

[11] N. Metropolis *et al.*, J. Chem. Phys. **21**, 1087 (1953).

[12] A. D. Simpson, Ph.D. thesis, University of Edinburgh, 1991.

[13] S. Duane, A. Kennedy, B. Pendleton, and D. Roweth, Phys. Lett. B **195**, 216 (1988).

[14] F. Fucito, E. Marinari, G. Parisi, and C. Rebbi, Nucl. Phys. B **180**, 369 (1981).

[15] S. Gottlieb *et al.*, Phys. Rev. D **35**, 2531 (1987).

[16] D. Callaway and A. Rahman, Phys. Rev. Lett. **49**, 613 (1982).

[17] J. Polonyi *et al.*, Phys. Rev. Lett. **53**, 644 (1984).

[18] S. Gupta and A. Irbäck and F. Karsch and B. Petersson, Phys. Lett. B **242**, 437 (1990).

[19] M. Creutz, Phys. Rev. D **38**, 1228 (1988).

[20] R. Gupta *et al.*, Phys. Rev. D **40**, 2072 (1989).

[21] M. Lüscher, Nucl. Phys. B **418**, 637 (1994).

[22] D. Young, in *Iterative Solutions of Large Linear Systems* (Academic Press, New York, 1971).

[23] J. Schwinger, Phys. Rev. D **16**, 2425 (1962).

[24] H. Dilger, Ph.D. thesis, DESY, 1991.

[25] J. Smit and J. Vink, Nucl. Phys. B **286**, 485 (1987).

[26] M. Atiyah and I. Singer, Ann. Math. **87**, 484 (1968).

[27] H. Dilger, `hep-lat/9408017` .

[28] M. Lüscher, Nucl. Phys. B (Proc. Suppl.) **42**, 49 (1995).

[29] B. Jegerlehner, Nucl. Phys. B (Proc. Suppl.) **42**, 879 (1995).

[30] P. Sawicki and J. Wosiek, Nucl. Phys. B (Proc. Suppl.) **42**, 932 (1995).

[31] C. Alexandrou *et al.*, `hep-lat/9506001` , to appear in Nucl. Phys. B.

[32] M. Peardon, Nucl. Phys. B (Proc. Suppl.) **42**, 891 (1995).

[33] A. Boriçi and P. de Forcrand, `hep-lat/9505021` , to appear in Nucl. Phys. B.

[34] A. Boriçi and P. de Forcrand, `hep-lat/9509080` , to appear in Proceedings Lattice '95 (Melbourne).

[35] P. de Forcrand, `hep-lat/9509082` , to appear in Proceedings Lattice '95 (Melbourne).

[36] G. Batrouni *et al.*, Phys. Rev. D **32**, 2735 (1985).

[37] J. Zinn-Justin, Nucl. Phys. B **275**, 135 (1986).

[38] W. Press *et al.*, *Numerical Recipies* (Cambridge University Press, Cambridge, 1988).

[39] K. Symanzik, Nucl. Phys. B **226**, 187 (1983).

[40] M. Alford *et al.*, `hep-lat/9007010` .

[41] M. Alford *et al.*, Nucl. Phys. B (Proc. Suppl.) **42**, 787 (1995).

[42] B. Berg and A. Billiore, Nucl. Phys. B **221**, 109 (1983).

[43] G. Lepage, Lectures given at TASI '93 (1993).

[44] G. Lepage and P. Mackenzie, Phys. Rev. D **48**, 2250 (1993).

[45] M. Lüscher and P. Weisz, Comm. Math. Phys. **97**, 59 (1985).

[46] P. Weisz and Wöhlert, Nucl. Phys. B **236**, 397 (1984).

[47] M. Lüscher and P. Weisz, Nucl. Phys. B **240**, 349 (1985).

[48] H. Fritzsch and M. Gell-Mann, Proc.16th Int. Conf. on High-Energy Physics **2**, (1972).

[49] G. Bali *et al.*, Phys. Lett. B **309**, 378 (1993).

[50] H. Chen, J. Sexton, and D. Weingarten, Nucl. Phys. B (Proc. Suppl.) **34**, 357 (1994).

[51] S. Collins *et al.*, `hep-lat/9509044` , to appear in Proceedings Lattice '95 (Melbourne).

[52] M. Teper, Phys. Lett. B **138**, 345 (1987).

[53] J. Lomont, *Applications of Finite Groups* (Academic Press, New York, 1959).

[54] C. Morningstar and M. Peardon, `hep-lat/9509069` , to appear in Proceedings Lattice '95 (Melbourne).

[55] A. Ferrenberg and R. Swendsen, Phys. Rev. Lett. **61**, 2635 (1988).

[56] M. Lüscher, Comm. Math. Phys. **104**, 177 (1986).

[57] C. Michael and M. Teper, Phys. Lett. B **206**, 299 (1988).

[58] C. Michael and M. Teper, Nucl. Phys. B **314**, 347 (1989).

[59] P. de Forcrand *et al.*, Phys. Lett. B **152**, 107 (1985).

[60] P. de Forcrand *et al.*, Phys. Lett. B **160**, 137 (1985).

[61] C. Morningstar, Private Communication .

[62] HEMCGC Collaboration, Nucl. Phys. B (Proc. Suppl.) **34**, 379 (1994).

[63] SESAM Collaboration, , to appear in Proceedings Lattice '95 (Melbourne).