MONTE CARLO METHODS IN LATTICE GAUGE THEORIES

Thesis by

Steve William Otto

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California.

1983

(Submitted May 2 ,1983)

Acknowledgements

I would like to thank my advisor, Geoffrey Fox, for his encouragement and the guidance he has given me these past few years. I also thank Rick Field for saving me from gravitational wave detection research and taking me under his wing.

I thank my collaborators, Mohit Randeria, Anthony Hey, Jan Ambjorn and Rajan Gupta for their friendly support and help. I especially thank Olivier Martin for putting up with me and teaching me much physics.

Finally, I thank my parents for their support in allowing me to get an education; as for my wife, Barbara, her warmth and love was most crucial in keeping me sane over the years. Thank you.

Abstract

In this work, we study various Monte Carlo methods for lattice gauge theories. The mass of the 0^+ glueball for SU(2) gauge theory in 4 dimensions is calculated. This computation was done on a prototype parallel processor and the implementation of gauge theories on this system is described in detail. Using an action of the purely Wilson form (trace of plaquette in the fundamental representation), we obtain results with high statistics. We conclude that these results are not consistent with scaling according to the continuum renormalization group. Using actions containing higher representations of the group, we search for one which is closer to the continuum limit. Our choice is based upon the phase structure of these extended theories and also upon the Migdal-Kadanoff approximation to the renormalization group on the lattice. We obtain the mass of the 0^+ glueball for this improved action and find that the mass divided by the square root of the string tension is a constant as the lattice spacing is varied. We conclude that scaling has set in and that this lattice theory is closer to the continuum limit than the simple Wilson version.

The other topic studied is the inclusion of dynamical fermions into Monte Carlo calculations via the pseudo fermion technique. Monte Carlo results obtained with this method are compared with those from an exact algorithm based on Gauss-Seidel inversion. We first apply the methods to the Schwinger model (QED in 1+1 dimensions) and show, in a coupling regime where the dynamical fermions have a nontrivial effect, that the mass gap is obtained with the correct value. After giving simple arguments explaining why the method works better than expected, we turn to a study of SU(3) in 4 dimensions (although on small lattices). Comparing with the exact algorithm, we again find encouraging agreement with the pseudo fermion technique. Evidence is given which shows that any systematic bias, associated with the breaking of the Markov process which generates the field configurations, is small.

Table of Contents

Introduction

The invention of lattice gauge theories [1] has profoundly deepened our understanding of quantum field theories. Besides providing an explicit ultraviolet cutoff, the lattice version of gauge theories allows one to attack the theory with methods beyond those of standard weak coupling perturbation theory (the Feynman graph expansion). Prominent among these methods are strong coupling expansions [2], Monte Carlo (numerical) estimates [3], and the renormalization group program (blocking or decimation of the fundamental degrees of freedom) [4].

The last few years have seen enormous progress in our understanding of the lattice theories themselves and this, in turn, has shed light on the non-perturbative aspects of QCD. Probably the most important result obtained so far is the rather strong numerical evidence (see Fig. 1.1) that SU(2) and SU(3) gauge theories in 4 dimensions are confining [5]. Much excitement has been generated by the "quenched" approximation to the path integral [6], in which internal quark loops are ignored. Though it is still unclear how accurate the approximation is, one can at least say that the results for the hadronic masses are encouraging. Another important area is the investigation of the spontaneous breakdown of chiral symmetry. In numerical calculations [7], the spontaneous breakdown is explicitly seen and it is shown that, for SU(3), the breakdown happens at length scales similar to the confinement scale.

In this work, we concentrate on Monte Carlo (numerical) techniques for lattice gauge theories. In Chapter I, an introduction and overview of lattice gauge

theories is given. We can, of course, only touch upon this rapidly growing subject and the topics chosen are biased toward what will be needed later. Excellent reviews of the subject have been given by Kogut [8].

Chapter II turns to the calculation of the glueball mass via Monte Carlo. Computational speed is a severe constraint on these calculations and we describe the Caltech effort at achieving large amounts of computer power through parallel processing. Glueball mass results from a prototype parallel processor are given. Though our results are almost identical to the results given by other groups, we find that they are inadequate and that continuum physics is not being accurately described in this calculation. One possible way of improving the calculation is to push to larger lattices via more powerful computers and in the future there is no doubt that this will be done. Another improvement is possible, however, and this is the subject of Chapter III. There is a freedom in the way the lattice theory is constructed, that is, there are different ways of discretizing the continuum Lagrangian. One can take advantage of this freedom and search for lattice actions which model continuum physics more accurately than the simplest lattice action. In Chapter III, we describe one method for finding such improved actions and we present results showing that it does give better results.

The inclusion of dynamical fermions into Monte Carlo calculations is a severe problem due to the generation of a non-local interaction between the gauge fields. Perhaps the most promising technique for including dynamical fermions is the pseudo fermion method, and this technique is studied in Chapter IV. Additional approximations to the pseudo fermion method, which make it a potentially very fast algorithm, are discussed. This approximate algorithm is then applied to QED in 1+1 dimensions and, on small lattices, SU(3) in 4 dimensions. In both of these calculations we compare with an exact (though very slow)

way of including the dynamical fermions. In both cases, we find the pseudo fermion algorithm to be fast and to give good agreement with the exact method.

Chapter I: Introduction to Lattice Gauge Theories

1. Basics

Let us briefly review some of the basic formalism of lattice gauge theories. The starting point is the Feynman path integral:

$$\int [d\varphi]\, e^{-S[\varphi]} \ . \tag{I.1.1}$$

Here, $\int [d\varphi]$ denotes functional integration over all possible configurations of the field, φ. We have set $\hbar = 1$ and have rotated to imaginary (Euclidean) time, $\tau = it$. The path integral associated with a quantum field theory plays the same role as the partition function for statistical mechanics. The action, S, is the analog of the Hamiltonian divided by temperature and physical observables, O, are obtained by averaging with the "Boltzmann factor":

$$< O[\varphi] > = \frac{\int [d\varphi]\, O[\varphi]\, e^{-S[\varphi]}}{\int [d\varphi]\, e^{-S[\varphi]}} \ . \tag{I.1.2}$$

The above functional integrals are rather formal objects; one way to explicitly make sense of them, and at the same time provide an ultraviolet cutoff for the theory, is to go to a spacetime lattice [9,10]:

$$\int [d\varphi]\, e^{-S[\varphi]} \to \int \cdots \int \prod_i d\varphi_i \, exp\left(-\sum_i L[\varphi] a^4\right) \ . \tag{I.1.3}$$

Spacetime has been discretized into cells of side a and the integrals on the right hand side are now ordinary integrals over the countably many variables φ_i, each of which represents the average of the field over cell i. For a simple, scalar field theory, derivatives appearing in the Lagrangian density, L, are replaced by finite

differences in the usual way. A slight problem arises for gauge theories, however. This is the fact that if one puts the theory onto the lattice in this straightforward way, then local gauge invariance will be broken for non-zero lattice spacing, a. One could ignore this and hope that exact gauge invariance is recovered in the limit $a \to 0$; after all, the lattice itself breaks Lorentz invariance, which we certainly expect to be recovered as $a \to 0$. We would like to break as few of the symmetries of the theory as possible, however, and since a simple way exists to make the lattice theory exactly gauge invariant, this is what is typically done.[1] Wilson [1] gave the following form of the lattice action for pure gauge, SU(N) theories.

Consider a four dimensional hypercubical lattice. Call the lines connecting adjacent sites "links"; associated with each link is an SU(N) matrix, $U_\mu(n)$, where n labels the site and is integer valued, and μ denotes the direction of the link originating from n. The path integral on the lattice is:

$$\int \prod_{n,\mu} dU_\mu(n)\, e^{-\beta S[U]} \quad . \tag{I.1.4}$$

β is $\frac{2N}{g^2}$, where g is the bare (lattice) coupling constant and $S[U]$ is a sum over all the elementary squares or "plaquettes" of the lattice,

$$S[U] = \sum_{plaqs} S_{plaq}$$

$$S_{plaq} = 1 - \frac{1}{2N} tr\left(U_\mu(n)\, U_\nu(n+\mu)\, U_\mu^{-1}(n+\nu)\, U_\nu^{-1}(n) + h.c. \right) \quad . \tag{I.1.5}$$

See Fig. 1.2 for a picture of this.

1. A study has been attempted for the case in which the lattice theory breaks local gauge invariance. See reference [20].

Matter fields are associated with the sites of the lattice. The U matrices are the SU(N) color rotations which a (colored) particle undergoes as it travels in the presence of the gauge field - if a quark moves from the site n to the site $n+\mu$, its wave function changes as,

$$\Psi(n) \rightarrow U_\mu(n)\,\Psi(n) \quad . \tag{I.1.6}$$

The inverse matrix, $U^{-1}=U^\dagger$, is associated with a move in the negative μ direction. The U's are related to the usual vector potential, A_μ, by:

$$U_\mu(n) = e^{-igaA_\mu(n)} \quad . \tag{I.1.7}$$

If one expands Eq. (I.1.5) for small a (assuming the fields have a smooth limit as $a \rightarrow 0$), one recovers the usual continuum, pure gauge action for SU(N) Yang-Mills:

$$-\beta \sum_{plaqs} S_{plaq} \rightarrow \frac{1}{2} \sum_{n,\mu\nu} tr\,(F_{\mu\nu}F^{\mu\nu})\,a^4 \tag{I.1.8}$$

$$\rightarrow \frac{1}{2}\int d^4x \;\; tr\,(F_{\mu\nu}F^{\mu\nu}) \quad ,$$

with,

$$F_{\mu\nu} = \partial_\mu A_\nu - \partial_\nu A_\mu - ig\,[A_\mu, A_\nu] \quad .$$

The lattice action, Eq. (I.1.5) is invariant under the transformation,

$$U_\mu(n) \rightarrow V(n)\,U_\mu(n)\,V^{-1}(n+\mu) \quad , \tag{I.1.9}$$

for all the link variables $U_\mu(n)$, where $V(n)$ is an arbitrary SU(N) matrix associated with the site n. This is the local gauge symmetry in the lattice formulation. The measure, $dU_\mu(n)$, in Eq. (I.1.4) is a group invariant measure, that is,

$$d(U'U_\mu(n)) = d(U_\mu(n)U') = dU_\mu(n) \quad , \tag{I.1.10}$$

for any SU(N) matrix, U'.

The reader used to continuum path integrals may be wondering about the lack of a gauge fixing term in the path integral on a lattice. In the continuum, such a term arises in the following way. Consider the Abelian case; the action is:

$$\frac{1}{4}(\partial_\mu A_\nu - \partial_\nu A_\mu)^2 = \frac{1}{2} A_\mu K_{\mu\nu} A_\nu \quad , \tag{I.1.11}$$

with,

$$K_{\mu\nu} = \partial^2 \delta_{\mu\nu} - \partial_\mu \partial_\nu \quad . \tag{I.1.12}$$

The operator, $K_{\mu\nu}$, is singular (and so, non-invertible). Any gauge configuration which is gauge equivalent to the zero configuration, i.e.,

$$A_\mu = \partial_\mu \Lambda \quad , \tag{I.1.13}$$

with Λ an arbitrary function, gives a zero contribution to the action:

$$K_{\mu\nu} \partial_\mu \Lambda = \partial_\mu \partial^2 \Lambda - \partial_\mu \partial^2 \Lambda = 0 \quad . \tag{I.1.14}$$

This is, of course, just the gauge invariance of the action. Since the action pays no attention to the longitudinal part of the field (the part proportional to $\partial_\mu \Lambda$), these are not damped out in the sum over all configurations and give an infinite contribution. The continuum path integral without gauge fixing will be ill-defined. The procedure, therefore, is to modify the path integral so as to integrate only over gauge non-equivalent configurations, with the introduction of the gauge fixing term (making $K_{\mu\nu}$ invertible) , along with the consequent appearance of "ghosts" [10].

The situation on the lattice is, however, different. If the gauge group is compact (as is usually the case) and if we have a finite number of sites, the integration over all gauge equivalent configurations, though not damped by the action, makes just a finite contribution to the path integral. Hence, gauge fixing is not necessary on the lattice.

The action of Eq. (I.1.5) is not unique. One can construct lattice actions consisting of traces of products of U's around closed loops larger than the elementary square. Higher representations (than the fundamental) of the group may also be included. If we again assume that the fields have a smooth limit as $a \to 0$, we can choose the coefficients of the various terms in the action so as to give the correct continuum action, as in Eq. (I.1.8). We will later take advantage of this freedom in the choice of the lattice action.

Before getting on to the main topic of this work, Monte Carlo methods for evaluating the path integral, we will briefly discuss the other two main methods: strong coupling expansions and renormalization transformations on the lattice. This is done for two reasons. First of all, it will give an overview of the techniques used in lattice gauge theories and will show where the present work fits in. Secondly, some of the results from these techniques will be used here - for instance, an approximate renormalization technique (Migdal-Kadanoff) will be used in Chapter III to find an improved lattice action for calculating glueball masses.

2. Strong Coupling Expansions

As an illustration of this technique, which will also give us an important result, we will calculate the expectation of Wilson loops for pure gauge U(1). The Wilson loop is the expectation value of the trace of a product of U matrices around some closed path, C:

$$W_C \equiv < \frac{1}{N} tr \prod_C U > \ . \tag{I.2.1}$$

Since C is closed, this is a gauge invariant quantity. What is the physical significance of this observable? Recall that the U matrices are related to the vector potential by,

$$U_\mu(n) = e^{iga A_\mu(n)} \ , \tag{I.2.2}$$

which can be written as,

$$U_\mu(n) = e^{i \int A_\mu J_\mu d^4x} \ , \tag{I.2.3}$$

with,

$$\vec{J}(x) = g \hat{\mu} a \, \delta^4(x - n) \ .$$

$\vec{J}(x)$ is the current for an infinitely massive charge. The Wilson loop measures the interaction of the gauge field with an external current loop. Suppose now that the loop is rectangular, of width R in the spatial direction, length T in the time direction, with $T \gg R$. Consider a time slice through the loop. At each time slice, there is a charge +1 at x=0, a charge -1 at x=R. Since these charges are static and we are measuring interaction energy, we have, for T large,

$$W_{R \times T} = e^{-T V(R)} \ . \tag{I.2.4}$$

$V(R)$ is the potential energy of the system. It includes both the interaction energy between the two charges (giving the force), and also the self-energy of each charge interacting with the field.

Let us now calculate this quantity for the U(1) gauge theory, for g large. The link variables are phase factors, $U_\mu(n) = e^{i\vartheta_\mu(n)}$, and the action (from Eq. (I.1.5)) is,

$$S = \frac{1}{g^2} \sum_{plaqs} \left[1 - \frac{1}{2} \left(e^{i \sum_{plaqs} \vartheta_\mu} + e^{-i \sum_{plaqs} \vartheta_\mu} \right) \right] . \qquad (I.2.5)$$

We are evaluating,

$$< exp(i \sum_C \vartheta_\mu) > = \frac{1}{Z} \int_0^{2\pi} \prod_{n,\mu} d\vartheta_\mu(n) \, exp(i \sum_C \vartheta_\mu) \, e^{-S} . \qquad (I.2.6)$$

Now consider the above for $g^2 \gg 1$. Expanding e^{-S} in powers of $\frac{1}{g^2}$, we find that the lowest orders of the expansion give zero. To see this, take an integral involving one of the angles of the contour, C. For low orders of the expansion we will just have,

$$\int_0^{2\pi} d\vartheta_\mu e^{i\vartheta_\mu} = 0 . \qquad (I.2.7)$$

"Exposed" phase factors give zero - they will contribute only if they are canceled by the phases in S to give,

$$\int_0^{2\pi} d\vartheta_\mu = 2\pi . \qquad (I.2.8)$$

Expand the exponential:

$$exp\left(\frac{1}{2g^2} \sum_p \left(e^{i\sum_p \vartheta_\mu} + e^{-i\sum_p \vartheta_\mu} \right) \right) = \prod_p \sum_n \frac{1}{n!} \left[\frac{1}{2g^2} \left(e^{i\sum_p \vartheta_\mu} + e^{-i\sum_p \vartheta_\mu} \right) \right]^n . \qquad (I.2.9)$$

Call those plaquettes which lie on the minimal spanning surface of C the "minimal spanning set." If, in Eq. (I.2.9), we pick out the $n=1$ term for each plaquette of this set (and the $n=0$ term for the others), we get the non-vanishing contribution with fewest powers of $\frac{1}{g^2}$. See Fig. 1.3. The outermost ring of plaquettes are included to cancel the exposed phase factors of the Wilson loop, but they, in turn, have exposed phase factors for a rectangle one unit smaller and

must be canceled, and so on, until the entire spanning surface is tiled. This gives,

$$< exp(i\sum_{C} \vartheta_{\mu}) > \sim \left[\frac{1}{g^2}\right]^{A} \quad , \tag{I.2.10}$$

where A is the area of the minimal spanning surface of C. Higher orders in this strong coupling expansion will correspond to spanning surfaces beyond the minimal one.

We have found that Wilson loops, for g large, fall off with an area law,

$$W_C \sim e^{-f(g^2)A} \quad . \tag{I.2.11}$$

Comparing with Eq. (I.2.4), we see that an area law fall off for $R \times T$ loops implies that

$$V(R) \sim R \quad , \tag{I.2.12}$$

that is, a linearly confining potential. Taken at face value, this is a crazy result - QED is not a confining theory! One can also perform a weak coupling expansion of the Wilson loop and find that the loop expectation then falls off as the perimeter of the loop (self energy), plus a term corresponding to the Coulomb potential. What happens between strong and weak coupling is that there is a phase transition (in 4 dimensions) at some finite value of g at which the loops change from area law decay to perimeter law decay [11].

For the case of SU(N), the non-abelian nature of the group leads to a more complicated strong coupling expansion [12]. For similar reasons to U(1), however, a minimal spanning set of plaquettes is again needed and so the Wilson loop has area law decay, Eq. (I.2.11). In contrast with U(1), it has been found that there is <u>no</u> phase transition between strong and weak coupling for SU(2) and

SU(3) gauge theories, using the action of Eq. (I.1.5). This is, therefore, evidence that in the weak coupling regime, Wilson loops also decay (asymptotically) with an area behavior and that the continuum theory is confining.

The result of Eq. (I.2.11) shows that as $g \to \infty$, the Wilson loop expectations all go to 0. As $g \to \infty$, the links are becoming uncorrelated with each other. We can define a correlation length, ξ, as,

$$W_C \sim e^{-\frac{A}{\xi^2}} , \tag{I.2.13}$$

so we see that as $g \to \infty$, $\xi \to 0$. For the lattice theory to have any relevance to the continuum, we want $\xi \to \infty$ in units of the lattice spacing, so that the system loses all "memory" of the lattice. Strong coupling expansions are an expansion in a region far from the continuum; for them to have quantitative significance for the continuum theory they must be carried to very high order (perhaps need to be summed to all orders).

3. Renormalization on the Lattice

The goal in lattice calculations is to work on large lattices, with couplings chosen so as to make the correlation length, ξ, very large, and to compute observables which are also very long range in character. In this way, one will be assured that the theory "forgot" the finite lattice spacing; one would be near the continuum limit. This ideal, though, is very hard to achieve. In strong coupling expansions, we have already mentioned that a large correlation length means that expansions will have to be carried to very high orders. For Monte Carlo techniques, ξ large means that large lattices must be used, and this, coupled with the critical slowing down of the speed with which the Monte Carlo travels through configuration space, will exhaust all present day computers.

One of the goals of the renormalization group program is to find a new, effective lattice theory which has a much smaller correlation length (in units of the lattice spacing) than the original theory, yet has the same behavior for large distance scale (measured in physical units) observables. In order to accomplish this, the effective theory must, in general, have complicated, non-local interactions.

The effective theory is obtained by integrating out high frequency degrees of freedom,

$$e^{S_{eff}[\varphi']} \equiv \int_{high\,\nu} [d\varphi] e^{S[\varphi]} \quad , \tag{I.3.1}$$

where the new field variables, φ', are some combination of the original variables, φ. A common choice is for φ' to be a local average of the φ variables in a small volume ("blocking"); another is for the φ' variables to be just a subset of the original variables (thinning of degrees of freedom, "decimation"). The new variables will be more sparse than before, that is, the lattice spacing of the new, effective theory will be some factor, λ, larger than the original. The effective theory is the renormalization of the original; its couplings are "running" couplings.

If we are working with a renormalizable field theory, the first few renormalization transformations will do nothing but change some of the couplings in S. S_{eff} will have the same functional form as S. Eventually, however, S_{eff} will change in structure and complicated, long range interactions will develop. If one could implement this program for QCD, for example, one would start with a theory of quarks and gluons which would be weakly coupled at some short distance scale due to asymptotic freedom. For the first few transformations, g would just be renormalized in accordance with the usual weak coupling β function. Eventually, the long range interactions will set in; we expect that these

would cause collective excitations of the degrees of freedom corresponding to "flux tubes", "strings", etc.. We would then have a model for the hadrons, derived from first principles. Continuing the renormalization process and redefining the field variables in a suitable way, we should eventually arrive at a theory of interacting baryons and mesons - i.e., nuclear physics!

The preceding discussion is the renormalization program in its most ambitious form. The transformations are very hard to implement accurately. A more realistic goal is to renormalize a lattice theory a few times, arriving at an effective theory with some non-local interactions. Suppose the effective theory is the result of N renormalization transformations, each of length scale change λ. Then, a correlation length of ξ in the effective theory will correspond to a correlation length of $(\lambda)^N \xi$ in the original. The effective theory will be, for a given correlation length, much "closer" to the continuum limit than the original one.

Renormalization can, in principle, be implemented exactly using Monte Carlo methods [13], the only limitation being computer time. Approximate, truncated analytic methods exist and we will illustrate one such method here for a simple model: Migdal-Kadanoff recursion applied to the 2 dimensional Ising model [14].

The 2 dimensional Ising model is pictured in Fig. 1.4a. On the sites are spins which can take on values ± 1 and the action is of nearest neighbor form with partition function,

$$Z = \sum_{\sigma=\pm 1} e^{\beta \sum_{<\sigma_1\sigma_2>} \sigma_1\sigma_2} \quad . \tag{I.3.2}$$

We wish to decimate the system by performing the sum in Z for every other column of spins - those labeled as σ_i in Fig. 1.4b. The variables labeled μ_i will

become the variables of the effective theory (φ' in Eq. (I.3.1)). In contrast to the 1 dimensional case, this sum cannot be done exactly. The problem is that the vertical bonds among the σ's of a column will generate non-local interactions among the μ's in the vertical direction. To proceed we must make an approximation. Assuming the spins are well correlated, we shift the vertical bonds between the σ spins over to those between the μ spins, obtaining the system shown in Fig. 1.4c. We now do the σ sum. Using,

$$e^{\beta\sigma_1\sigma_2} = \cosh\beta\,(1+\sigma_1\sigma_2\tanh\beta)\ ,$$

$$\sum_{\sigma=\pm1}(1+k\,\mu_1\sigma)(1+k\,\mu_2\sigma) = 2(1+k^2\mu_1\mu_2)\ , \tag{I.3.3}$$

we get:

$$\sum_{\sigma=\pm1} e^{\beta\mu_1\sigma}e^{\beta\mu_2\sigma} = 2\cosh^2\beta\exp[\tanh^{-1}(\tanh^2\beta)\mu_1\mu_2]\ . \tag{I.3.4}$$

Absorbing the irrelevant constant, we see that this first renormalization transform has changed the couplings to,

$$\beta_x \to \tanh^{-1}[\tanh^2\beta] \tag{I.3.5}$$

$$\beta_y \to 2\beta\ ,$$

where β_x,β_y are, respectively, the couplings in the x and y directions. Now perform the same transformation on the system but in the other direction. At the end, we will have:

$$\beta_x = 2\tanh^{-1}[\tanh^2\beta] \tag{I.3.6}$$

$$\beta_y = \tanh^{-1}[\tanh^2(2\beta)]\ .$$

Unfortunately, these are not equal - our effective theory is no longer isotropic.

This must be an artefact of our approximations since the original system was isotropic. Therefore, average the couplings to regain an isotropic theory:

$$\beta_{eff} \equiv \frac{1}{2}(\beta_x + \beta_y) \quad . \tag{I.3.7}$$

The above was for a scale change of $\lambda = 2$. The result for any integer λ is easily generalized to,

$$\beta_x^\lambda = \lambda \tanh^{-1}[\tanh^\lambda \beta] \tag{I.3.8}$$

$$\beta_y^\lambda = \tanh^{-1}[\tanh^\lambda(\lambda\beta)] \quad .$$

We wish to find the change in coupling for small changes in the length scale (the "beta" function). To do this, assume we can use Eq. (I.3.8) for λ non-integer and, in particular, near 1. This can be interpreted as a procedure in which only some of the vertical columns of σ's are integrated out, and then the coupling defined as the average of the couplings in that direction (we want the system to stay homogeneous). Once again, we are assuming the system to be well correlated. We have also numerically compared Eq. (I.3.8), for non-integer λ, with the result of such an averaging procedure and find that the two compare quite closely (5%) over the β range for which it will be used. Expanding Eq. (I.3.8) for $\lambda = 1 + \Delta$, Δ small, the isotropy is recovered and we find,

$$\beta^\lambda = \beta + \Delta[\beta + \sinh\beta \cosh\beta \ln(\tanh\beta)] \quad . \tag{I.3.9}$$

The function,

$$\beta + \sinh\beta \cosh\beta \ln(\tanh\beta) \quad , \tag{I.3.10}$$

governs the change of couplings with scale changes. It has a zero at $\beta = \beta_c \approx .436$, to be compared with the exact result of .4407 . This zero corresponds to a fixed

point - under renormalization the action reproduces itself and so the correlation length is infinite. This is part of the reason why the Migdal-Kadanoff method is able to find the location of the critical point so well - the approximations become good as $\xi \to \infty$. The quality of the agreement with the exact result is also somewhat accidental.

What about the non-local interactions which we threw away? Martinelli and Parisi [14] show how one can systematically improve the method by including some of the non-local terms generated by the spin decimation. They get improved agreement with the exact result for both the location of the critical point and a critical exponent at this point.

The Migdal-Kadanoff method for gauge theories is similar to the spin model application. As we will later discuss, the decimation involves the integration of planes of link variables; the bond shifting of the Ising model becomes the shifting of plaquettes. For the gauge theory case, we will use an effective action which has several couplings, so the form of the action will actually change under renormalization, in contrast to the above Ising model approximation.

4. The Monte Carlo Method

In this method the ensemble of all possible configurations is sampled, numerically, via an algorithm which produces a biased, random walk through configuration space. Observables are then found, from Eq. (I.1.2), as simple averages of "measurements" made in the ensemble. These estimates, since they are statistical in nature, converge as $\frac{1}{\sqrt{N}}$, where N is the number of configurations sampled. The exponent, $e^{-\beta S[\varphi]}$, implies that the sampling of configurations cannot be done in a purely random way. $S[\varphi]$ consists of a sum of terms over the entire lattice and therefore grows as the volume of the space-time lattice. This means that $e^{-\beta S[\varphi]}$ is a function which is incredibly peaked

about those configurations which minimize $S[\varphi]$. If we just randomly sampled field configurations, we would never hit those configurations which give the major contribution to the path integral.

What is done, instead, is to pick the field configurations in a random way, but with the probability of each configuration appearing proportional to the exponential factor,

$$P[\varphi] \sim e^{-\beta S[\varphi]} \quad . \tag{I.4.1}$$

Observables then become simple averages over this biased ensemble:

$$\frac{1}{Z}\int [d\varphi]\, O[\varphi]\, e^{-\beta S[\varphi]} = \frac{1}{N}\sum_{\varphi_i} O[\varphi_i] \quad . \tag{I.4.2}$$

In this way, the Monte Carlo explores only the important part of configuration space since the configurations which minimize S will appear often in the ensemble.

The problem of distributing configurations according to Eq. (I.4.1) is equivalent to the numerical simulation of a thermodynamic system in thermal equilibrium. The probability corresponds to the Boltzmann factor, with β as the inverse temperature and S as the Hamiltonian. The generation of the ensemble is accomplished in the following way. Start with some arbitrary configuration. Typical choices are totally ordered configurations ("cold" start) or totally random ones ("hot" start). This configuration is then changed into a new one according to a set of transition probabilities $P_{i\to i+1}$, i labeling the configuration. In this way, a random walk through configuration space is constructed. It is called a "Markov chain" because the choice of the ith configuration depends only on the $i-1$th configuration and not the $i-2$th, $i-3$th, etc. . This has practical importance in that only one configuration need be stored in the computer at any given time.

A sufficient (but not necessary) constraint on the $P_{i\to i+1}$, so that the ensemble satisfies Eq. (I.4.1), is that of "detailed balance". Let n_i be the number of times that configuration i appears in the ensemble. Then if,

$$n_i P_{i\to k} = n_k P_{k\to i} \quad , \tag{I.4.3}$$

the system will reach thermal equilibrium, and if we choose the transition probabilities so that:

$$\frac{P_{k\to i}}{P_{i\to k}} = \frac{n_i}{n_k} = \frac{e^{-\beta S_i}}{e^{-\beta S_k}} = e^{-\beta(S_i - S_k)} \quad , \tag{I.4.4}$$

then the correct ensemble will be obtained, since the right hand side is just the Boltzmann factor.

A particularly simple and powerful way of satisfying detailed balance is the method of Metropolis, et.al. [15]. Suppose we have some configuration, k, and we take some field variable and change it by some random amount to get a new configuration, i (i differs from k only at a single site). We compute the change in the action, $S_i - S_k$. If $S_i - S_k < 0$, the change in the field variable is accepted; if $S_i - S_k > 0$, the change is accepted with probability $e^{-\beta(S_i - S_k)}$. The algorithm then goes on to repeat the same procedure on the next field variable, eventually sweeping through the entire lattice. This method satisfies Eq. (I.4.3). If $S_i - S_k < 0$ then $P_{k\to i} = 1$ and $P_{i\to k} = e^{-\beta(S_k - S_i)}$; if $S_i - S_k > 0$ then $P_{k\to i} = e^{-\beta(S_i - S_k)}$ and $P_{i\to k} = 1$.

As long as the transition probabilities satisfy detailed balance, we are guaranteed [16] that, asymptotically (in the Markov chain), the configurations generated by this procedure will be distributed according to the Boltzmann factors. The time it takes for this to happen, from the arbitrary starting configuration, is called the thermalization time.

One further requirement which must be satisfied in order for the Markov chain to give the correct ensemble is that of ergodicity, that is, all configurations must be reachable. This can be a problem if the system is at a phase transition so that distinct phases can exist. For example, in some spin models rotational symmetry can be spontaneously broken - the spins all point in some particular direction. If one is in this phase and starts the Markov chain with all spins "up," they will stay "up." On any finite lattice, such states are only metastable - all configurations will be explored (it takes only a finite number of "spin flips" to change the overall magnetization), though it may take a long time.

Another ergodicity problem is that certain configurations may be topologically disallowed by the boundary conditions chosen for the finite lattice. For example, if one chooses periodic boundary conditions on a 2 dimensional lattice (i.e. a torus) for the planar spin model, then the total vorticity is rigorously zero. Configurations with non-zero vortex number won't be present in the ensemble. This is not, however, really a problem. If the lattice is sufficiently large, local quantities such as the density, distribution, etc., of vortices, will take on their correct values. We would be working in the canonical ensemble instead of the grand-canonical ensemble of the theory.

A few practical remarks concerning the Metropolis Monte Carlo method. The calculation of the change in the action, $S_i - S_k$, involves, since the action is local, the calculation of just a few terms involving field variables nearby the one which has been moved.[2] Since the ith configuration evolves out of the $i-1$th, these configurations will be highly correlated. Measurements made in them will not be statistically independent and for this reason, one typically only bothers to measure observables once every 10-100 sweeps (a "sweep" being defined as

2. This no longer is true when dynamical fermions are included. This case will be discussed in Chapter IV.

the application of the Metropolis procedure to the entire lattice). When estimating error bars, one must be careful to take such correlations into account.

Monte Carlo methods have been quite successful. They are not limited to strong or weak couplings and can, in principle (given adequate computer time!), compute any observable to any desired degree of accuracy.

5. Fermions

As was mentioned earlier, matter fields live on the sites of the lattice. For scalar fields, no problems appear when derivatives of the Lagrangian are replaced by finite differences in the usual way. For spin 1/2 fermions, however, a problem does arise when this is done. Unphysical modes, which do not go away in the continuum limit, appear.

Consider the Dirac equation in 1+1 dimensions. The Dirac field is a two component spinor satisfying,

$$\partial_t \psi = \alpha \partial_x \psi \ . \tag{I.5.1}$$

Put the system onto a spatial lattice: $x = n a$. Then, replacing ∂_x by a (symmetric) finite difference, we get,

$$\partial_t \psi(n) = \frac{\alpha}{2a}(\psi(n+1) - \psi(n-1)) \ . \tag{I.5.2}$$

Expand ψ in terms of $\psi_\pm$, where $\psi_\pm$ are eigenvectors of α:

$$\alpha \psi_\pm = \pm \psi_\pm \ . \tag{I.5.3}$$

Trying a plane wave solution, $e^{-i(kx-Et)}\psi_\pm$, then leads to the dispersion relation,

$$E = \pm \frac{1}{a}\sin(ka) \ , \quad -\frac{\pi}{a} \leq k \leq \frac{\pi}{a} \ , \tag{I.5.4}$$

while the dispersion relation in the continuum is,

$$E = \pm k \ , \quad -\infty < k < \infty \ . \tag{I.5.5}$$

As we take $a \to 0$, the modes near $k = 0$ are the ones we are interested in - they are long wavelength (in lattice units), and the energy of these modes becomes correct: $\frac{\sin(ka)}{a} \to k$ as $a \to 0$. The modes at $k = \pm \frac{\pi}{a}$, howver, stay finite in energy as $a \to 0$, even though their momentum goes to infinity. Since their energy is small, they will, for example, appear in a Monte Carlo simulation of the system. This is the problem of unphysical modes in the lattice formulation of spin 1/2 particles.

One needs to put the spin 1/2 field onto the lattice in such a way so as to prevent the appearance of unphysical modes. One way to do this is the approach of Susskind [17]. He splits the Dirac spinor, putting upper components on the even sites, lower components on the odd sites. This doubles the size of the unit cell and has the effect of halving the size of the Brillouin zone: k now runs from $-\frac{\pi}{2a}$ to $\frac{\pi}{2a}$. This removes the problem at $k = \pm \frac{\pi}{a}$, and the unphysical modes do not appear. This method has the disadvantage that, in higher dimensions, though all unphysical modes (E finite, k infinite) are removed, extra flavors appear. Starting with a Lagrangian which, in the continuum, describes only one fermion, one gets, on the lattice in 4 dimensions, 4 fermion species.

The other approach to removing unphysical modes is due to Wilson [18]. In this method, all the components of the Dirac spinor are at each site. An extra term is added to the lattice Lagrangian which does not effect the modes at $k = 0$, but raises the energy of those near $k = \pm \frac{\pi}{a}$, so that, in the continuum limit their energy goes to infinity and they decouple from the system. This method has the

disadvantage that chiral symmetry is broken for non-zero lattice spacing; in the Susskind approach, some of the chiral symmetry remains, even for a non zero. However, the Wilson approach does not have species doubling problems.

Since we will be using the Wilson approach to putting fermions on the lattice, let us now explicitly give the lattice Lagrangian. The action on the lattice is:

$$S[U,\psi,\bar{\psi}] = S_{gauge}[U] + \sum_{ij} \bar{\psi}_i \, Q_{ij} \, \psi_j \quad . \tag{I.5.6}$$

Q is given by

$$Q_{x\alpha A,y\beta B} = \delta_{xy} \, \delta_{\alpha\beta} \, \delta_{AB} - \kappa \, M_{x\alpha A,y\beta B} \tag{I.5.7}$$

with

$$M_{x\alpha A,y\beta B} = (1+\gamma^{\mu})_{\alpha\beta} (U_{x,\hat{\mu}})_{AB} \, \delta_{x,y-\hat{\mu}} + (1-\gamma^{\mu})_{\alpha\beta} (U^{\dagger}_{x-\hat{\mu},\hat{\mu}})_{AB} \, \delta_{x,y+\hat{\mu}} \tag{I.5.8}$$

$$\{\gamma^{\mu},\gamma^{\nu}\} = 2\,\delta^{\mu\nu} \quad ,$$

x,y are lattice sites, α,β are Dirac indices, and A,B are color indices. The projection operators $1+\gamma^{\mu}, 1-\gamma^{\mu}$ act to remove the spurious fermion modes from the lattice formulation. In the continuum limit κ is related to the mass, m, the lattice spacing a, and the dimension d, by:

$$\kappa = \frac{1}{2am + 2d} \quad . \tag{I.5.9}$$

Since in the Wilson formulation there is no chiral symmetry to prevent the appearance of mass terms under renormalization, κ is renormalized, and $\kappa = \frac{1}{2d}$ does not necessarily mean that the bare mass, m_0, is zero. For finite a, κ must be fine-tuned so as to produce the correct result for some observable such as the mass of the pion.

The combined fermion-gauge theory is described by the path integral,

$$\int [dU][d\bar{\psi}]d[d\psi]e^{-S} \quad . \tag{I.5.10}$$

The fermion variables, ψ, are anticommuting (Grassman) numbers so as to enforce Pauli exclusion. A direct Monte Carlo evaluation of the above path integral is possible in 1+1 dimensions [19]. By going to the number representation for the fermions, the Pauli principle can be simply satisfied - a site is either occupied or unoccupied. Though this approach works well in 1+1 dimensions, it breaks down in higher dimensions, where the anticommuting variables cause negative signs to appear. This would mean that observables would be found through huge cancelations and so the method is unreliable.

Another approach is to simply integrate out the fermion field. For most interesting theories, the fermion field appears as a bilinear and the field can be integrated out using [21],

$$\int [d\bar{\psi}][d\psi]e^{\bar{\psi}Q\psi} = \det Q \quad . \tag{I.5.11}$$

Only the gauge fields are left, but they now interact via the non-local determinant (in addition to the usual pure gauge action). To apply the Metropolis method of updating the gauge fields, we need to evaluate the ratio of two such determinants. If a particular gauge link is moved:

$$U \rightarrow U + \delta U \, , \tag{I.5.12}$$

then Q will change:

$$Q \rightarrow Q + \delta Q \, , \tag{I.5.13}$$

and we need to compute

$$\frac{det(Q+\delta Q)}{det(Q)} = det(1 + Q^{-1}\delta Q) \, . \tag{I.5.14}$$

δQ is a known, simple matrix. What is needed is an efficient way to find $Q^{-1}[U]$, the fermion propagator in an external gauge field, which depends on all of the U variables of the lattice. We will investigate one possible method in Chapter IV.

References

[1] K. G. Wilson, Phys. Rev. D14, 2455 (1974).

A. M. Polyakov, Phys. Lett. B59, 82 (1975).

F. Wegner, J. Math. Phys. 12, 2259 (1971).

[2] K. G. Wilson, Phys. Rev. D14, 2455 (1974).

T. Banks, S. Raby, L. Susskind, J. Kogut, D. Jones, P. Scharbach, D. Sinclair, Phys. Rev. D15, 1111 (1977).

M. Creutz, Rev. Mod. Phys. 50, 561, (1978).

[3] M. Creutz, Phys. Rev. D21, 2308 (1980).

K. G. Wilson, Cargese lecture notes (1979).

[4] K. G. Wilson and J. Kogut, Phys. Rep. 12, 75 (1974).

K. G. Wilson, Cargese lecture notes (1979).

[5] M. Creutz, Phys. Rev. Lett. 45, 313 (1980).

M. Creutz, K. J. M. Moriarty, Phys. Rev. D26, 2166 (1982).

[6] D. Weingarten, Phys. Lett. 109B, 57 (1982).

H. Hamber, G. Parisi, Phys. Rev. Lett. 47, 1792 (1981).

E. Marinari, G. Parisi, C. Rebbi, Phys. Rev. Lett. 47, 1795 (1981).

[7] J. Kogut, M. Stone, H. W. Wyld, J. Shigemitsu, S. H. Shenker, D. K. Sinclair, Phys. Rev. Lett. 48, 1140 (1982).

J. Kogut, M. Stone, H. W. Wyld, W. R. Gibbs, J. Shigemitsu, S. H. Shenker, D. K. Sinclair, Illinois preprint, ILL-(TH)-82-39.

[8] J. Kogut, Rev. Mod. Phys. 51, 659 (1979).

J. Kogut, Les Houches and St. Andrews lectures, 1982, Illinois preprint, ILL-(TH)-82-46.

[9] R. Feynman, A. Hibbs, *Quantum Mechanics and Path Integrals*, (McGraw-Hill, 1965).

[10] E. S. Abers, B. W. Lee, Physics Reports 9, 1 (1973).

[11] T. A. DeGrand, D. Toussaint, Phys. Rev. D22, 2478 (1980).

[12] M. Creutz, Rev. Mod. Phys. 50, 561, (1978).

[13] K. G. Wilson, Cargese lecture notes (1979).

[14] G. Martinelli, G. Parisi, Nucl. Phys. B180[FS2], 201 (1981).

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

[16] p. 5, *Monte Carlo Methods in Statisical Physics*, edited by K. Binder (Springer-Verlag, New York, 1979).

[17] L. Susskind, Phys. Rev. D16, 3031 (1977).

[18] K. G. Wilson, in *New Phenomena in Subnuclear Physics*, edited by A. Zichichi (Plenum Press, New York, 1977).

[19] J. E. Hirsch, D. J. Scalapino, R. L. Sugar, R. Blankenbecler, Phys. Rev. Lett. 47, 1628 (1981).

O. Martin, S. Otto, Nucl. Phys. B203, 297 (1982).

[20] A. Patrascioiu, E. Seiler, I. O. Stamatescu, Phys. Lett. 107B, 364 (1981).

[21] p. 214-218, *Field Theory, A Modern Primer*, P. Ramond, (Benjamin,1981).

Figure Captions

[1.1]The string tension for SU(2) (1.1a) and SU(3) (1.1b) gauge theories. It is the lower envelope of the data which is the string tension; the reliability is to be judged by how well the envelope follows the continuum renormalization group result (the sets of 3 solid lines). These figures are from reference [5].

[1.2]The plaquette corresponding to Eq. (I.1.5). n is the site in the lower left corner and the inverse matrix is used when traveling in the negative direction.

[1.3]The strong coupling expansion for a Wilson loop. The outer curve is the contour C, the inner squares are the plaquettes from the expansion of the action which cancel the exposed phase factors.

[1.4](a) The 2 dimensional Ising model. On the sites are spins taking on the values ± 1. (b) The columns of spins labeled σ are being integrated out (decimated), the columns labeled μ become the variables of the effective theory. (c) The lattice after the bond shifting.

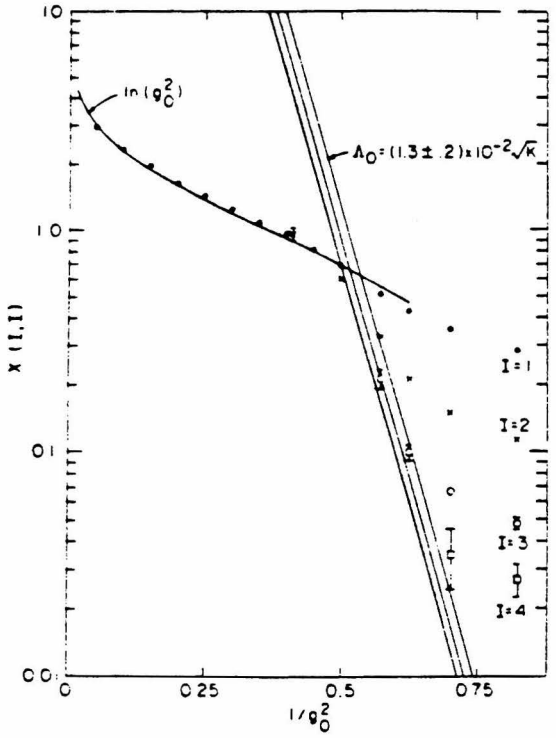


Fig. 1.1a

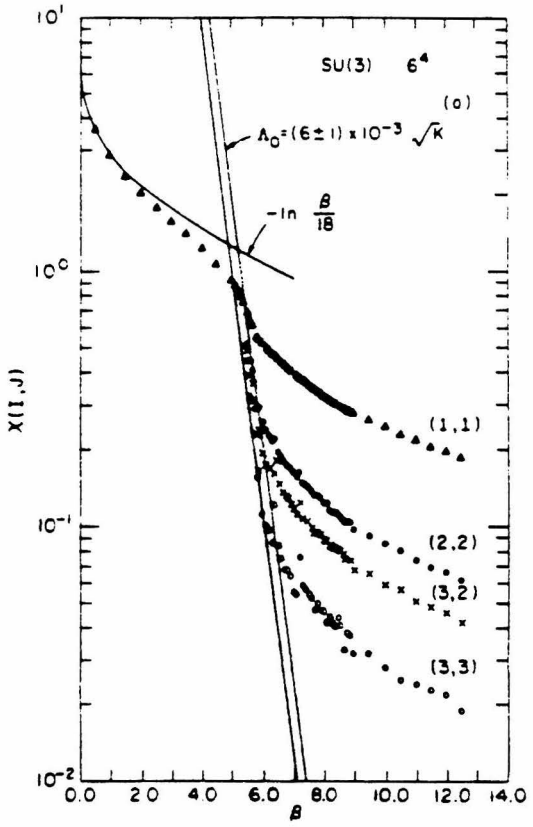


Fig. 1.1b

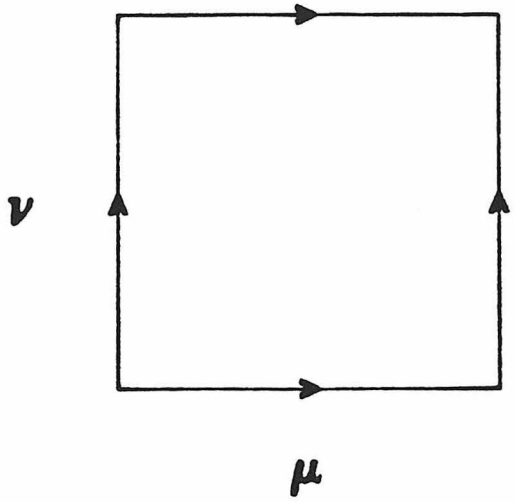


Fig. 1.2

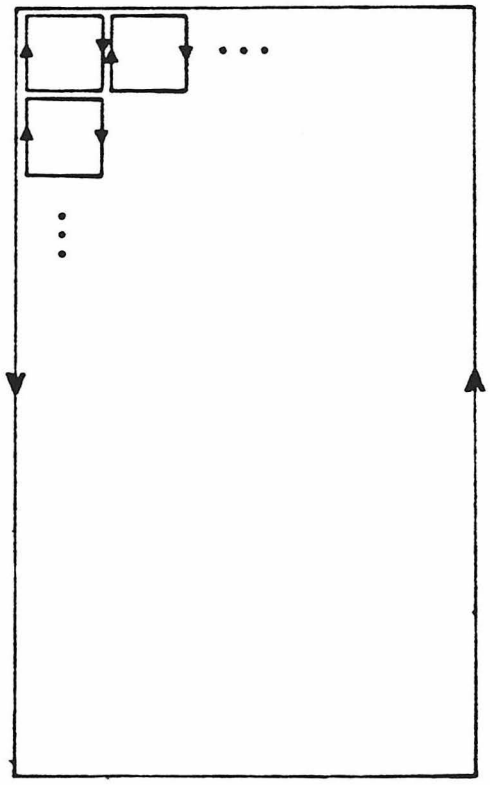
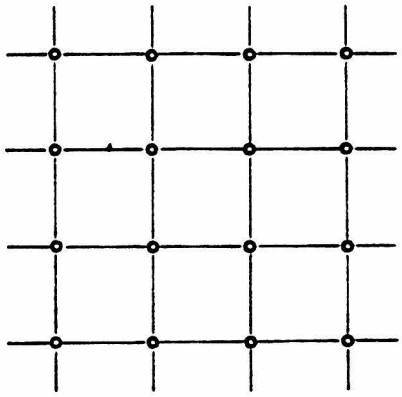
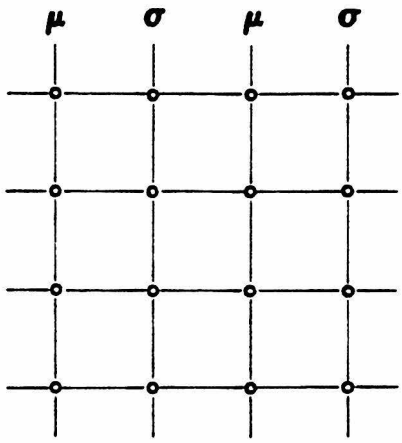


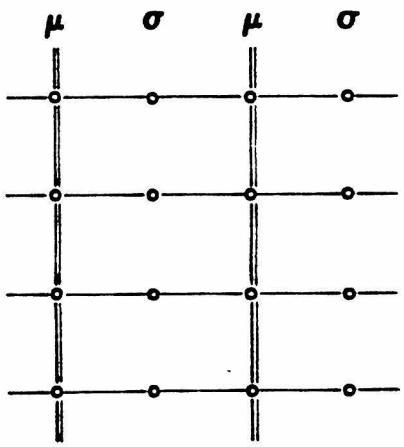
Fig. 1.3



(a)



(b)



(c)

Fig. 1.4

Chapter II: The Glueball Mass on an Array of Computers

Introduction

In this chapter, we will report on some calculations of the mass of the 0^+ glueball (a bound state composed mostly of gluonic degrees of freedom) in SU(2) pure gauge theory. Some of this work has been previously described in [1]. The relevance of our calculation to the real 0^+ glueball mass is unclear. First of all, the effect of dynamical quarks (internal loops) is being set to zero. This "quenched" approximation can be partly justified on both experimental grounds: the apparent validity of the OZI rule (quark pairs are hard to create); and on theoretical grounds: the success that such quenched calculations have had in explaining some features of the flavor non-singlet part of the hadronic spectrum. Secondly, we are using the group SU(2) instead of the correct one, SU(3). This is done for practical reasons - the Monte Carlo calculation using SU(2) proceeds approximately an order of magnitude faster than the SU(3) version. Instead of attempting to make physical predictions, we are taking a more skeptical attitude and are trying to judge the reliability of such Monte Carlo calculations. In this way, the SU(2) glueball calculation should be looked upon as a model of what the "ultimate" calculation will require.

Section 1 will discuss methods for calculating masses in lattice gauge theories. The need for the computation of exponentially small numbers is pointed out. The difficulty of achieving reliable estimates of such small effects leads to two conclusions: huge amounts of computer power are needed, and an improved lattice calculation must be found, perhaps along the lines of the renormalization transformation methods as outlined in Chapter I. Section 2 will

describe the Caltech effort at achieving large amounts of computational power, involving parallel processing on arrays of microprocessors, and the implementation of pure gauge theories on such machines. The results of the glueball calculation, using the usual Wilson form of the action (Eq. (I.1.5)), on a prototype parallel processor (an array of 4 computers), are given in section 3. These results are in agreement with those given by other groups. The fact that the data do not convincingly scale (i.e. not depend on the lattice spacing used) is pointed out. Improved calculations are considered in the next chapter.

1. Masses in Lattice Gauge Theories

Estimates of the mass spectrum of a theory can be obtained through the use of the asymptotic decay of 2 point correlation functions of operators which have a non-zero overlap with the state in question. Recall that the path integral can be written as [2],

$$Z = tr\, e^{-TH} = \sum_k < k \mid e^{-TH} \mid k > \ , \qquad \text{(II.1.1)}$$

where T is a time corresponding to the inverse temperature of the system, and $\mid k >$ is some complete set of states. The usual path integral is recovered by inserting complete sets of states in every Δt interval:

$$Z = \sum_{k,l,\cdots} < k \mid e^{-\Delta t H} \mid l >< l \mid e^{-\Delta t H} \mid n > \cdots < a \mid e^{-\Delta t H} \mid k > \ . \qquad \text{(II.1.2)}$$

Now consider the 2 point correlation of some operator O, from time 0 to time τ. As $T \to \infty$, the only state in Eq. (II.1.1) which will contribute is the ground state (the rest are exponentially down). Using this, and taking the complete set of states to be energy eigenstates, we find,

$$< O(\tau)\, O(0) >_{T \to \infty} = \sum_n \mid < n \mid O \mid 0 > \mid^2 e^{-\tau (E_n - E_0)} \ . \qquad \text{(II.1.3)}$$

Taking the "connected" 2 point function, defined as,

$$< O(\tau)\, O(0) > - < O(\tau) >< O(0) > \ , \qquad \text{(II.1.4)}$$

we find,

$$< O(\tau)\, O(0) > - < O(\tau) >< O(0) > = \sum_{n \neq 0} | < n \mid O \mid 0 > |^2 \, e^{-\tau (E_n - E_0)} \ . \qquad \text{(II.1.5)}$$

That is, only the states higher than the ground state (the vacuum) contribute to the connected function. Measuring the exponential decay of connected correlations of various operators therefore gives us a way to find the spectrum of the theory. For finite temperature (T finite in Eq. (II.1.1)), which corresponds to the actual case in Monte Carlo calculations, Eq. (II.1.5) is modified by the addition of an additional term corresponding to propagation from 0 to τ the "other" way around the lattice (the lattice is periodic in time at finite temperature). More details on this effect can be found in [3].

By picking O to have particular quantum numbers (i.e. be an eigenstate of various symmetries of the Hamiltonian), we can isolate particular states and find their masses reliably. An important example is momentum. If O is not chosen to be an operator of definite momentum, then there will be a continuum of states contributing to the right hand side of Eq. (II.1.5), with energies given by $E^2=p^2+m^2$. On the lattice it is not an actual continuum since p can take on only a discrete set of values. However, the mixing with higher states will be severe, and since the lattice dispersion relation between E,p,m is not known, it becomes difficult to extract a mass reliably. Typically, O is chosen to be an operator of momentum zero.

With the amount of statistics available in present day Monte Carlos, it has not been possible to extract more than the lowest lying mass for each choice of spin and parity (for the glueball sector). This is found by looking at the large τ

decay of Eq. (II.1.5); one needs to go to large τ so that contributions from higher states (with the same spin, parity, etc.) are negligible. The calculation can be improved by taking advantage of the remaining freedom in the choice of O. If we could find an O which exactly created the state (100% overlap) we were interested in, then the contributions from higher states would be zero (the states are guaranteed to be orthogonal). This forms the basis of an optimized 2 point correlation method, or, as it is commonly called, "Monte Carlo Variational Method" (MCVM) [4]. The usual variational method is based upon:

$$E \leq \frac{<\varphi \mid H \mid \varphi>}{<\varphi \mid \varphi>} . \qquad \text{(II.1.6)}$$

A minimum is then taken over φ's to get the best estimate of E. This is equivalent to,

$$e^{-E\Delta t} \geq \frac{<\varphi \mid e^{-H\Delta t} \mid \varphi>}{<\varphi \mid \varphi>} , \qquad \text{(II.1.7)}$$

so that maximizing the right hand side gives the best estimate of E. This maximization is exactly the same as, in the 2 point correlation method, maximizing the "signal" or overlap with the lowest state.

The MCVM method is a combination of variational and 2 point correlation techniques. There are two extreme ways of using the method. The first would consist of using a very poor choice of O and extracting the mass from large τ values, the second would be to work very hard at finding an O with high overlap and then getting the mass estimate from the first time slice, τ=a. Since calculating a complicated operator would cost much computer time, as would running long enough to get the 2 point function for τ very large, the optimum method, in terms of computer time needed, probably lies somewhere between the two extremes. An additional bonus of the MCVM method is that not only is the mass found, but some information about the actual wavefunctional of the state is

obtained through the operator O.

We have seen that mass estimates require the computation of an exponentially small signal. For glueballs and other flavor singlets (of spin 0^+), this computation is difficult for an additional reason. These states can mix with the vacuum, i.e. $< O >$ is not zero, causing the exponentially small signal to be computed as the difference of large, fluctuating quantities (Eq. (II.1.5)). There has been little progress in finding a more stable method to computing glueball masses [17] and this remains one of the most difficult computations in terms of the amount of statistics required.

In the next section, we will describe the effort underway at Caltech to achieve large amounts of computer power.

2. Parallel Processing and Lattice Gauge Theories

Fundamental limits on computational speed and feature size in VLSI technologies suggest that significant increases in performance will come not from pushing current designs yet further, but instead from new computer architectures utilizing many computers in parallel: concurrent processing [5]. A simple design for such a computer is a "Homogeneous Machine:" a regular array of (independent) processors with a small number of interconnections per processor [6]. Such a machine is actually being built at Caltech and consists of a 4×4×4 array of microprocessors wired as a 3 dimensional (periodic) cube and has a total CPU power of 10 VAX 11/780's. We have found that the Homogeneous Machine design is suited for use on many computationally intensive problems in the physical sciences such as partial differential equations, matrix inversion, and fast fourier transforms [7].

The locality of the actions used in Monte Carlo simulations of gauge theories implies that these computations can be concurrently processed in a

straightforward way. In the standard Metropolis procedure, for instance, the change in the action due to the move of a link variable, ΔS, involves only nearby link variables coupled to the one in question through plaquettes. This means that sets of decoupled links (links which nowhere appear in the same plaquette) can be simultaneously updated via the standard procedure, and the gauge field configurations will be generated according to the correct distribution, i.e. according to Eq. (I.4.1). This is because for each update, the ΔS's computed will not feel the effects of the other links being updated at the same time (they are decoupled) and so they will be exactly the same as for the usual sequential update algorithm. The simultaneous update of N decoupled links is exactly equivalent to the sequential update, in arbitrary order, of these same N links.

The maximum number of decoupled links which can exist on the lattice at any given time is $\frac{1}{4}N_{tot}$, where N_{tot} is the total number of links of the lattice. The structure of this maximal set of decoupled links is illustrated in Fig. 2.1, for 2,3 and 4 dimensions.

Beyond the simultaneous update of all the elements of a set of decoupled links, one could simultaneously update more than one link in a given plaquette. One must be careful, however, to use a procedure which generates the correct multi-link probability distribution, and not merely the product of the simple one-link distributions. As an example, suppose we wanted to simultaneously update 2 links, both of which are in the same plaquette. Using the Metropolis method, one could proceed as follows. Move each of the link matrices: $U_1 \rightarrow U_1+\delta U_1$, $U_2 \rightarrow U_2+\delta U_2$. Then compute the total change in the action,

$$\Delta S = S(U_1+\delta U_1, U_2+\delta U_2) - S(U_1, U_2) \quad . \tag{II.2.1}$$

Now, if ΔS is < 0, or if $e^{-\beta\Delta S} > r$ with r a random number $\in[0,1]$, accept <u>both</u> moves, δU_1 and δU_2. Otherwise, reject <u>both</u> moves, and return the matrices to

U_1 and U_2 . The above method satisfies the constraint of detailed balance and can of course be generalized to more than 2 link matrices. Currently, we are not running in this multi-link update mode - we are simultaneously updating sets of decoupled links.

Pure gauge theories have been programmed on the four element prototype array which was built to test the hardware and software for the 64 element microprocessor array currently being constructed. The four element prototype is configured as shown in Fig. 2.2. The processors labeled (0-3) are the nodes of the array which have bidirectional communication paths shown by solid lines. Each of these processors is based on the Intel 8086/8087 microprocessor. This microcomputer[1] has about $\frac{1}{8}$th the power of a VAX 11/780 in typical scientific computing. Each microprocessor board has 128k bytes of memory and 6 bidirectional data channels which allow communication with other processors in the array.

The machine labeled IH is the intermediate host, which is also an Intel 8086/8087 based processor. This processor functions as the controller of the array and also as a data buffer between the VAX and the array.

The application and system programming for the Homogeneous Machine is all performed on a VAX with a cross compiler producing 8086 code which is then loaded into the array. Currently, all our programs are written in the high level language C.

We will now discuss some of the details of 4 dimensional pure gauge algorithms on a 4-node concurrent processor. This machine illustrates all of the essential features of larger machines. In particular, the algorithms developed for this machine will run on the 64 node machine with a few minor modifications.

1. For an 8086-8087 processor running at 5 mhz.

The lattice is divided up among the computers so that neighboring variables of the lattice are either in the same node or in neighboring nodes. For the 2×2 square, the 4 dimensional lattice is divided up in 2 of its dimensions, the other 2 dimensions are "squashed" into the processors: if the total lattice is 4×4×4×4, each node stores a 2×2×4×4 subcell of the lattice.

To illustrate how the algorithm for gauge theories works on a homogeneous machine we will outline the steps required to update a link residing in one of the subcells of the lattice. To be definite, suppose we are updating the link labeled A in Fig. 2.3a. For the sake of simplicity only a two dimensional example is discussed. To update link A, the matrices $BC^{-1}D^{-1}$ and $E^{-1}F^{-1}G$ must be constructed and passed to processor 1. As will become clear, the algorithm is written in such a way so as to keep the processors synchronized - that is, the situation is actually as shown in Fig. 2.3b. At the same time that processor 1 is updating link A_1, processor 0 is updating link A_0, and so on. The corresponding matrices, $B_i C_i^{-1} D_i^{-1}$ and $E_i^{-1} F_i^{-1} G_i$, must be constructed and passed to processor i. The first step in the algorithm is for the B matrices to be exchanged between processors 0 and 2, 1 and 3. All such communications are done via a polled-mailbox scheme. Taking the 1,3 exchange as an example, processor 3 sends the matrix B_1 to the "mailbox" (an internal buffer) of processor 1 across the bidirectional channel "vchan" (vertical channel). After putting the matrix in the mailbox, the "flag" of the mailbox is set, indicating to processor 1 that the mailbox is ready to be read. Processor 1 likewise sends matrix B_3 to the mailbox of processor 3 and sets the flag. Each processor then polls their mailbox - checking the flag to see if there is something there to be read. If the flag indicates the mailbox is full, the processor empties the buffer, resets the flag to the "empty" position and proceeds to its next instruction. If the flag indicates that the mailbox is still empty, the processor effectively halts: it polls the flag

indefinitely until the mailbox is filled. This is how the processors are kept synchronized - if processor 3 is lagging behind 1, when 1 gets to the instruction to read its mailbox, it will stop at that point in its instruction sequence until 3 writes to it.

Once this B exchange is finished, the matrices B_i reside in the same processors as C_i and D_i, so the products $B_i C_i^{-1} D_i^{-1}$ are formed and are ready to be used for the update. The products $E_i^{-1} F_i^{-1} G_i$ are a bit more difficult since the matrices E_i are in the next-nearest neighbor processor to the one containing A_i. In order to minimize communication time, we adopt the strategy of passing E_1 to processor 0, forming the product $E_1^{-1} F_1^{-1} D_1$ in processor 0, and passing only the resultant product to processor 1. The same is, of course, simultaneously done for E_0, E_2 and E_3. The matrices are again passed via the polled-mailbox scheme as described above for the matrices B_i. Once all this is accomplished, the processors all update their A_i and then proceed to the next link, staying in step due to the nature of the communications software.

The above may seem complicated to implement but in reality it is not. Once the fundamental matrix exchange subroutines are written, all that is required beyond the usual Metropolis update algorithm is a few logical "if" statements which are needed to detect if the matrices B_i, etc. need to be communicated from a neighboring processor. In fact, the entire coding and debugging (starting from debugged VAX code) for SU(2) in 4 dimensions required only about 20 hours of time: a modest investment for the computer power which is gained.

We have implemented pure gauge SU(2) on the 4 node machine, using the 120-element icosahedral subgroup method to speed up the computation. This method speeds up our calculation by a factor of about 3. It has been shown [8] that this discrete approximation to SU(2) is sufficiently fine grained so as not

to affect results in the region of couplings in which we will work. Since we work with a finite subgroup, only integers labeling the members of the subgroup need to be communicated between the processors; actual matrices are not passed. Each node of the machine stores the entire group multiplication table of 14,400 integers. As a partial check of the correctness of the algorithm on the 4 node machine, we have verified that the usual average plaquette results were obtained - see Fig. 2.4.

Since the speed of each node is $\frac{1}{8}$th VAX, one would naively expect the performance of the 4 node machine to be that of $\frac{1}{2}$ of a VAX 11/780. This is, of course, degraded by the communications overhead present in a homogeneous machine, but not in a normal, sequential computer. This overhead was measured by timing the program, and then timing a version of the program in which all communications were done twice, doubling the communication overhead. The difference in these timings then gave the time spent in interprocessor communication. The results are shown in Fig. 2.5 for various lattice sizes, and are given in terms of percentage of total time spent in communications. Since this percentage is governed by the surface area to volume ratio of the subcells residing in each node, one expects the overhead to be worse for smaller subcells. This is apparent in Fig. 2.5, where the worst case of a 2×2×4×4 subcell gave a communications percentage of 25%. The fact that the overhead grows slowly as the subcell is made smaller and that the "worst case" overhead is still a reasonably small fraction is important. We want to add more nodes (the 64 node machine) and run on lattices of $8^3\times16$, for example, so the subcells in each node will always be fairly small. The 25% figure means that the performance of these larger machines will not be severely degraded by communications overhead, at least for this icosahedral version of SU(2). The overheads given can actually be improved upon by at least a factor of 2; we have not yet fully optimized our

communications software.

One may worry that in a more realistic gauge theory, i.e. SU(3), one will have to pass full 3×3 complex matrices between the nodes so that the communication overhead may become quite large. This turns out not to be the case, however. The reason is that although the communication time does indeed grow, the computation time per matrix communication grows even faster. For example, if the size of the matrix is N, matrix multiplication grows as N^3, while the communication time goes as N^2. We have actually implemented SU(3) and find that the communications overhead is, in fact, smaller: for the worst case 2×2×4×4 subcell, we find an overhead of 18% .

3. The Glueball Mass

The MCVM method, as described in Section 1, was used for calculating the mass of the 0^+ glueball. The operator O was taken to be a linear combination of gauge invariant operators,

$$O = \sum_i \alpha_i W_i \quad . \tag{II.3.1}$$

The W_i are combinations of Wilson loops chosen so as to excite states of definite spin, parity and momentum. The mass is,

$$m = \lim_{\tau\to\infty} m(\tau) \ , \tag{II.3.2}$$

with,

$$m(\tau) \equiv \ln \frac{\Gamma(\tau-1)}{\Gamma(\tau)} \ , \tag{II.3.3}$$

and,

$$\Gamma(\tau) \equiv \left\langle (O(\tau)-\langle O\rangle)(O(0)-\langle O\rangle) \right\rangle \ . \tag{II.3.4}$$

The signal, $\frac{\Gamma(1)}{\Gamma(0)}$, was maximized as a function of the parameters α_i. One could try to maximize other combinations of Γ's, such as $\frac{\Gamma(2)}{\Gamma(0)}$ or $\frac{\Gamma(2)}{\Gamma(1)}$. Due to the statistical noise in the data, such procedures turned out to be unreliable - if one of the correlation functions ($\sim< W_i(\tau)\, W_j(0) >$) happens to fluctuate upwards at $\tau=2$, the maximization algorithm would then choose the corresponding coefficient very large, giving bad results for the other time slice estimates, $m(\tau)$. Similarly, including more operators, W_i, in the maximization does not always improve the mass estimate. $m(1)$ is, of course, always lowered by the addition of more operators, since the minimization is done on $m(1)$. This does not necessarily hold for $m(2), m(3), \cdots$, however, if the operator is too noisy. For a given amount of statistics, only a limited number of operators should be included in the mass estimates. More operators should be added only if the the amount of statistics gathered on all the correlations is correspondingly increased.

We now present the results for the SU(2) 0^+ glueball mass. The action used is that given in Eq. (I.1.5) with the U matrices in the fundamental representation of SU(2). We have worked at six values of the coupling, β=2.0, 2.1, 2.15, 2.2, 2.25 and 2.3 on a 4×4×4×8 lattice. We have collected data with very large statistics; for our most ambitious data point (β=2.3) a total of 250,000 sweeps was generated. Roughly two-thirds of the time was spent in update and one-third in measurements. The entire computation, for all values of the coupling, took approximately 1000 hours on the 2×2 Homogeneous Machine, which is equivalent to 400 hours on the VAX 11/780.

The five operators used in the variational calculation are shown in Fig. 2.6. These are the simple plaquette, all three operators of perimeter six and the 2×2 planer loop. The calculation of such observables on the Homogeneous Machine is

nontrivial. Rather than give the details here, an explanation of the algorithm used is given in Appendix 1.

To construct an operator of spin 0, all spatial rotations of the loops are added together with coefficient 1. The lattice has only cubic symmetry, so this combination of loops also excites spin 4 states. We will assume that the spin 4 mass is above that of the spin 0 and interpret our lowest state as spin 0. All the spin 0 combinations of loops on a time slice are then added together, with coefficient 1, to form a translationally invariant, zero momentum state. Using the loops of Fig. 2.6, the parity of our state is +1. For example, the inversion of loop c in Fig. 2.6 gives just a rotation of that loop and is already included with relative coefficient 1. To get a 0^- state, one could use a loop such as the one shown in Fig. 2.7. The inversion of this loop cannot be reached by a rotation. Hence, using this loop and its inverse, with relative phase -1, and then taking all rotations (with +1 phase), will yield a 0^- state.

We chose our lattice to be only 4 sites on a side, in the spatial directions, for reasons of statistics. Since we construct a zero momentum state, the 2 point correlation function is a correlation between sums of operators over entire time slices. This function is a large scale observable and this causes it to move very slowly as the Monte Carlo proceeds, that is, it is very correlated with itself from sweep to sweep. This effect becomes much more severe as the time slices are increased in size, hence, we were limited to 4^3 time slices. We do not believe this lattice is too small for our calculation. We shall see that the correlation length for our range of couplings is 1.0 - 1.5, therefore, our lattice is approximately 3 times the size of the relevant length scale.

The glueball mass (times the lattice spacing) as a function of the coupling is plotted in Fig. 2.8. The error bars are statistical and take into account the

sweep to sweep correlations in the data.[2] The different symbols of the figure correspond to the different time slice estimates of the mass, i.e. $m(\tau)$ in Eq. (II.3.3). Since the lowest state is isolated as $\tau \to \infty$, we expect the $m(\tau)$ to be equal for τ sufficiently large, signaling that the isolation has indeed happened. Therefore, it is the lower envelope of the points in Fig. 2.8 which are the mass predictions, as a function of β (or g).

The lines drawn in the figure correspond to the prediction of the continuum renormalization group in two loop perturbation theory. The numbers which come out of a Monte Carlo calculation are masses in units of the inverse lattice spacing, $\frac{1}{a}$. This scale can be related to a more usual scale, such as Gev. The coupling constant, g, can be thought of as the (running) coupling at the scale of the lattice spacing: $g = g(a)$. For g sufficiently small, changes in a are related to changes in g by the usual, continuum renormalization group expression. Including two loops, one finds [9] that the quantity,

$$\frac{1}{a}(\beta_0 g^2)^{-\frac{\beta_1}{2\beta_0^2}} exp(-\frac{1}{2\beta_0 g^2}) \quad , \tag{II.3.5}$$

($\beta_0 = \frac{11}{24\pi^2}$ and $\beta_1 = \frac{17}{96\pi^4}$ for SU(2)) is a constant, which is conventionally called the "lattice mass scale", Λ_L. This is related to the more conventional scale, Λ_{MOM}, by

$$\Lambda_{MOM} = 57.4\Lambda_L \quad (SU(2)) \quad . \tag{II.3.6}$$

---

2. sweep to sweep correlations among the data have been taken into account in the estimation of the error through the use of the formula,

$$\sigma^2 = \frac{<A^2>-<A>^2}{N}[\,1 + 2\sum_{p=1}^{\infty}\frac{<A_i A_{i+p}>-<A>^2}{<A^2>-<A>^2}\,] \quad ,$$

where the subscript on the observable A labels sweep number.

The number, 57.4, represents the difference between the regularization schemes of the lattice cutoff and the usual continuum methods, such as dimensional regularization [10]. The lines in Fig. 2.8 are $a\Lambda_L$, times various constants. If the lower envelope of the data coincides with one of these lines, it means that a is moving with g in accordance with the continuum renormalization group and that the lattice spacing is small enough to be modeling continuum physics accurately.

We note that our results are almost identical to those of Berg, Billoire and Rebbi [11], though we have somewhat more statistics. In particular, the very low values for $m(3)$ at β= 2.15, 2.2, a prominent feature of our data, are also present in the data of [11]. See Fig. 2.9. This, coupled with the fact that both $m(3)$ values are approximately 2 standard deviations lower than the $m(2)$ values, leads us to conclude that they are a real feature and not merely a statistical fluctuation.

If we restrict ourselves to just the τ=2 mass estimates, as is done in [12], the data do seem to be consistent with scaling according to the continuum renormalization group, and we can quote a value, in physical units, for the glueball mass. In terms of Λ_L, we get,

$$m(0^+) = (170 \pm 20)\Lambda_L \quad ,$$

in agreement with [11].

Including the τ=3 mass estimates, we do not think that the data are consistent with scaling. The 0^+ state is clearly not isolated at τ=2 for β=2.15, 2.2, and the true values (the lower envelope) are far below the $(170 \pm 20)\Lambda_L$ lines. Our conviction that this is the case is strengthened by an examination of the specific heat in this same region of couplings. The specific heat is given by,

$$C = \beta^2 \frac{\partial}{\partial \beta} < W_p > = \beta^2 \sum_{p'} [\, < W_p \, W_{p'} > - < W_p >^2] \quad , \qquad \text{(II.3.7)}$$

where W_p is the trace of the pth plaquette. Since C is expressible in terms of the plaquette-plaquette correlation function, summed over the lattice volume, it is clear that a high value of C corresponds to a low value for $m(0^+)$ and vice-versa. Lautrup and Nauenberg [13] found C, via Monte Carlo, for the β range we are considering. Their results are shown in Fig. 2.10. A large peak in C is found at β=2.2 . We interpret our low mass values at 2.15, 2.2 as being associated with the peak in the specific heat.

This peak is a lattice artifact and has nothing to do with the physical, continuum theory. Using a lattice action with an adjoint representation plaquette in addition to the usual, fundamental representation plaquette, it was found [14] that there is an actual phase transition near the region in which we are working. The action used was,

$$S_{plaq} = \beta_F[\,1 - \frac{1}{2}Tr_F(U_{plaq})] + \beta_A[\,1 - \frac{1}{3}Tr_A(U_{plaq})] \quad . \qquad \text{(II.3.8)}$$

As discussed in Chapter I, this action gives the same continuum theory, as $\beta_F,\beta_A \to \infty$, as Eq. (I.1.5). In the β_F,β_A plane, the phase structure shown in Fig. 2.11 is found. We see that the region containing the strange behavior in the specific heat, β=2.2, is the linear extrapolation of a nearby line of phase transitions. The peak is a "shadow" of nearby, nontrivial behavior in the lattice theory.

In view of this, coupled with our direct evidence of anomalously low mass values, we conclude that scaling behavior is <u>not</u> seen for the range $2.1 \leq \beta \leq 2.3$. This conclusion is contrary to those given in [11,12]. To reliably find the mass of the 0^+ glueball, using the fundamental representation action of Eq. (I.1.5), one needs to work at $\beta \geq 2.3$. This requires more CPU power than we presently have available in our 4 node, prototype computer.

The situation is very similar for SU(3). In this case, there is also a specific heat peak in the relevant coupling range. The extended plane phase diagram is very similar to the SU(2) diagram, see Fig. 2.12. In fact, the line of critical points comes closer to the β_F axis (for SU(4) it actually crosses [15]). Berg and Billoire [16] also note that mass values seem to be too low, again, this occurs at the location of the peak in the specific heat. Though they conclude that the 0^+ data do scale, we feel that the nearby phase transition has not been adequately considered.

To improve the reliability of these glueball calculations, two things can be done. The first is simply to push to smaller lattice spacings ($\beta \geq 2.3$) with faster computers. The second is to search for lattice actions which model the large distance behavior of the continuum theory more accurately. This is the topic of the next chapter.

References

[1] E. Brooks, G. Fox, S. Otto, M. Randeria, W. Athas, E. DeBenedictis, M. Newton, C. Seitz, Caltech preprint, CALT-68-985, (1983), to be published in Nuclear Physics B.

[2] R. Feynman and A. Hibbs, *Quantum Mechanics and Path Integrals*, (McGraw-Hill, New York, 1965).

D. Gross, R. Pisarski, L. Yaffe, Rev. Mod. Phys. 53, 43 (1981).

[3] M. Creutz, B. Freedman, Ann. Phys. 132, 427 (1981).

[4] K. G. Wilson, closing remarks at the Abingdon Lattice Meeting (March, 1981).

K. Ishikawa, G. Schierholz, M. Teper, Phys. Lett. 110B, 399 (1982).

B. Berg, A. Billoire, C. Rebbi, Ann. Phys. 142, 185 (1982).

[5] C. Seitz, Proceedings, Conference on Advanced Research in VLSI, MIT, (1982).

R. Levine, Scientific American, Jan. 1982, p.118.

[6] E. Brooks et al., Caltech preprint, CALT-68-867 (1982).

[7] G. C. Fox, Caltech preprint, CALT-68-986 (1982).

[8] C. Rebbi, Phys. Rev. D21, 3350 (1980).

D. Petcher, D. Weingarten, Phys. Rev. D22, 2465 (1980).

G. Bhanot, C. Rebbi, Nucl. Phys. B180[FS2], 469 (1981).

[9] M. Creutz, Phys. Rev. Lett. 45, 313 (1980).

[10] A. Hasenfratz, P. Hasenfratz, Phys. Lett. 93B, 165 (1980).

[11] B. Berg, A. Billoire, C. Rebbi, Ann. Phys. 142, 185 (1982).

[12] K. Ishikawa, G. Schierholz, M. Teper, DESY preprint, DESY 83-004 (1983).

[13] B. Lautrup, M. Nauenberg, Phys. Rev. Lett. 45, 1755 (1980).

[14] G. Bhanot, M. Creutz, Phys. Rev. D24, 3212 (1981).

[15] M. Creutz, Phys. Rev. Lett. 46, 1441 (1981).

[16] B. Berg, A. Billoire, DESY preprint, DESY 82-079 (1982).

[17] C. Rebbi, review talk given at the ITP Functional Integrals Workshop, Santa Barbara, August, 1982.

Figure Captions

[2.1]Maximal set of decoupled links (bold lines) in (a) 2 dimensions, (b) 3 dimensions, (c) 4 dimensions.

[2.2]Schematic diagram of 4 node Homogeneous Machine.

[2.3]Example of pure gauge update to illustrate inter-processor communication.

[2.4]The average plaquette as a function of coupling for SU(2) gauge theory in 4 dimensions.

[2.5]Communication overhead as a function of subcell size (L, where each processor stores a $L\times L\times 4\times 4$ subcell of the lattice) for SU(2).

[2.6]Operators used in the variational calculation of the glueball wavefunction.

[2.7]Example of a loop which can be used to construct a parity negative state. The parity (i.e. inversion through the origin) transform of this loop cannot be reached by a rotation of the loop.

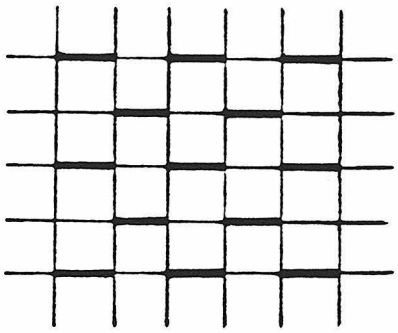
[2.8]Mass of the 0^+ glueball as a function of coupling for SU(2) gauge theory.

[2.9]The data of Berg, Billoire and Rebbi for the 0^+ glueball. This figure is from B. Berg, CERN preprint, .TH.3327-CERN, (1982).

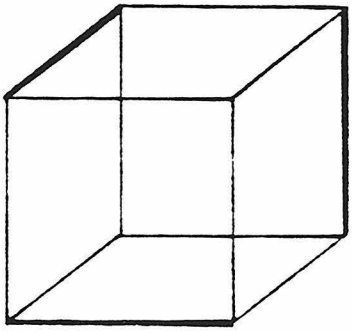
[2.10]The specific heat, C, as a function of β. The curves are strong and weak coupling expansions. From reference [13].

[2.11]The phase diagram of SU(2) in the extended β_F,β_A plane. The plotted points are the locations of first order phase transitions. From reference [14].

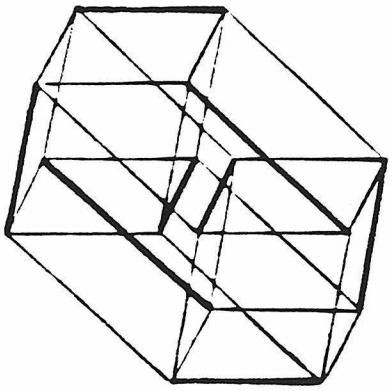
[2.12]The phase diagram of SU(3) in the extended plane. The points on the line AB are the locations of the first order phase transitions. The points on the bold line correspond to transitions occuring due to the use of a finite subgroup, S(648), and, for our purposes, can be ignored. This figure is from G. Bhanot, Phys. Lett. 108B, 337 (1982).



(a)



(b)



(c)

Fig. 2.1

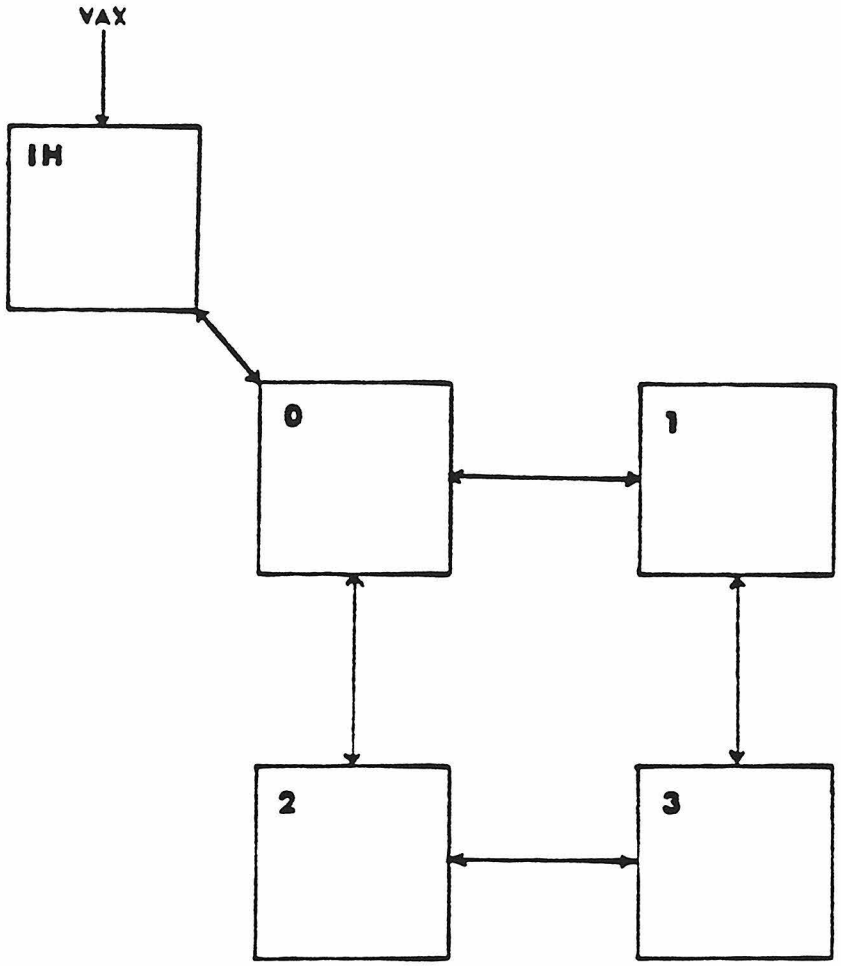
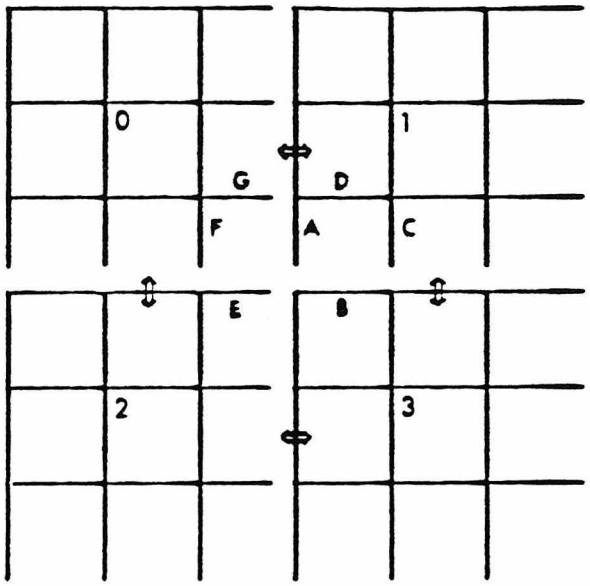
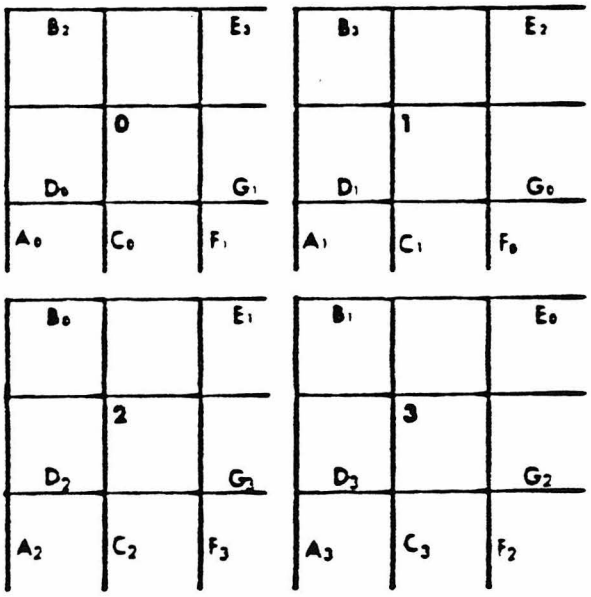


Fig. 2.2


0
1
G
D
F
A
C
E
B
2
3

(a)


B2
E3
0
D0
G1
A0
C0
F1
B3
E2
1
D1
G0
A1
C1
F0
B0
E1
2
D2
G3
A2
C2
F3
B1
E0
3
D3
G2
A3
C3
F2

(b)

Fig. 2.3

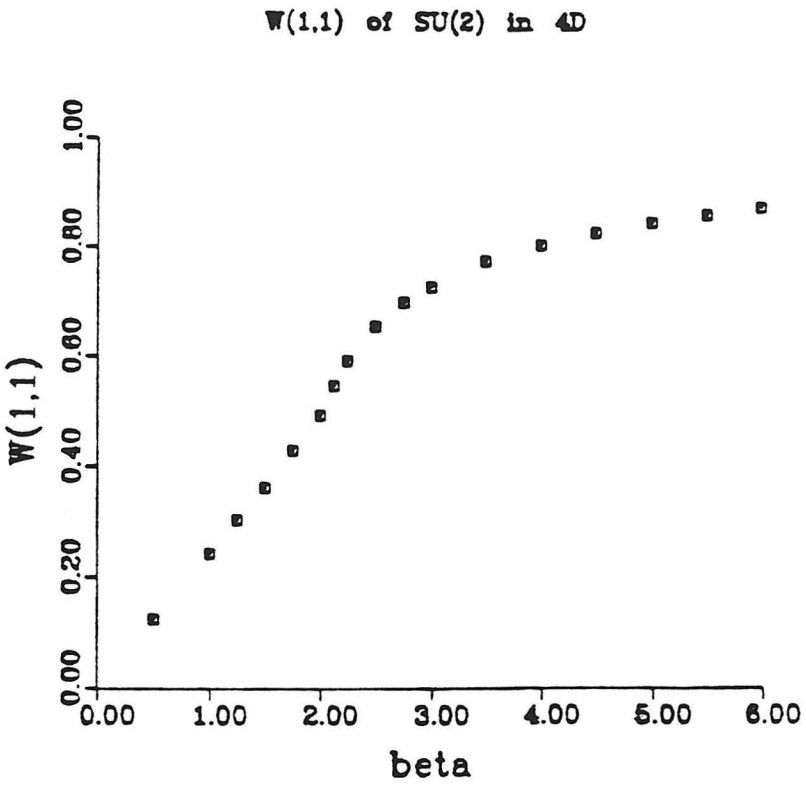

W(1,1) of SU(2) in 4D
W(1,1)
0.00
0.20
0.40
0.60
0.80
1.00
0.00
1.00
2.00
3.00
4.00
5.00
6.00
beta

Fig. 2.4

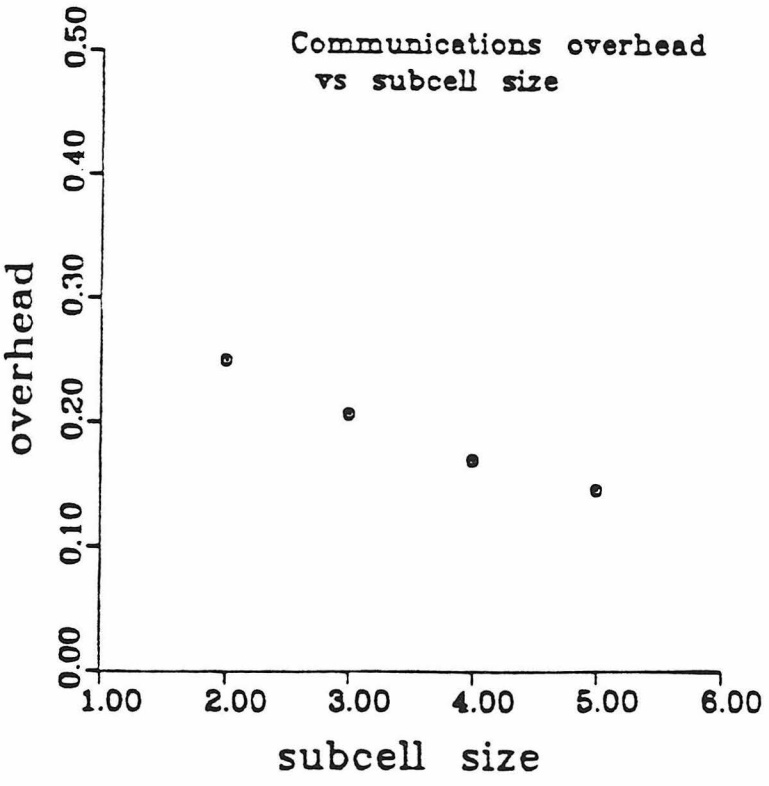

Communications overhead
vs subcell size
overhead
0.00
0.10
0.20
0.30
0.40
0.50
1.00
2.00
3.00
4.00
5.00
6.00
subcell size

Fig. 2.5

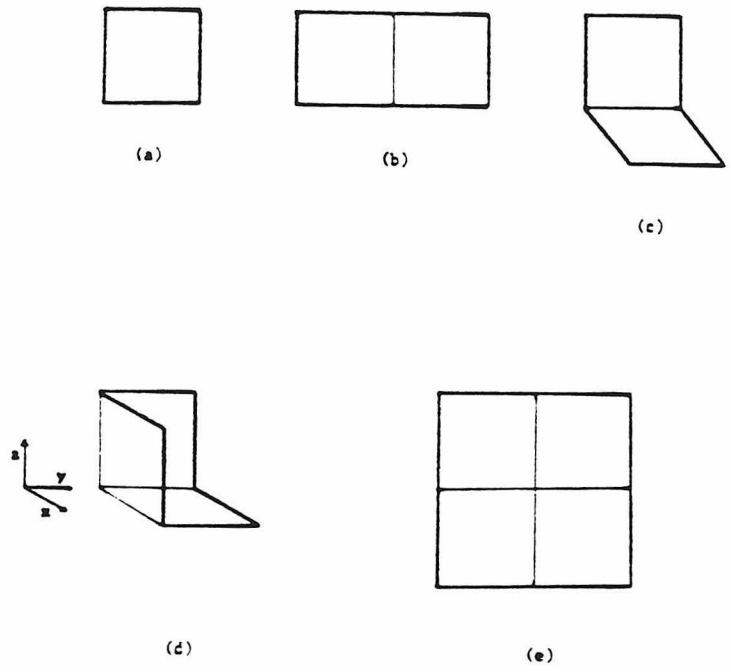


Fig. 2.6

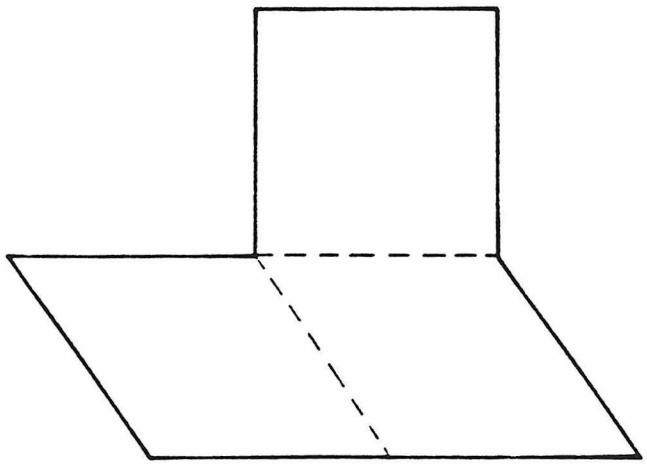


Fig. 2.7

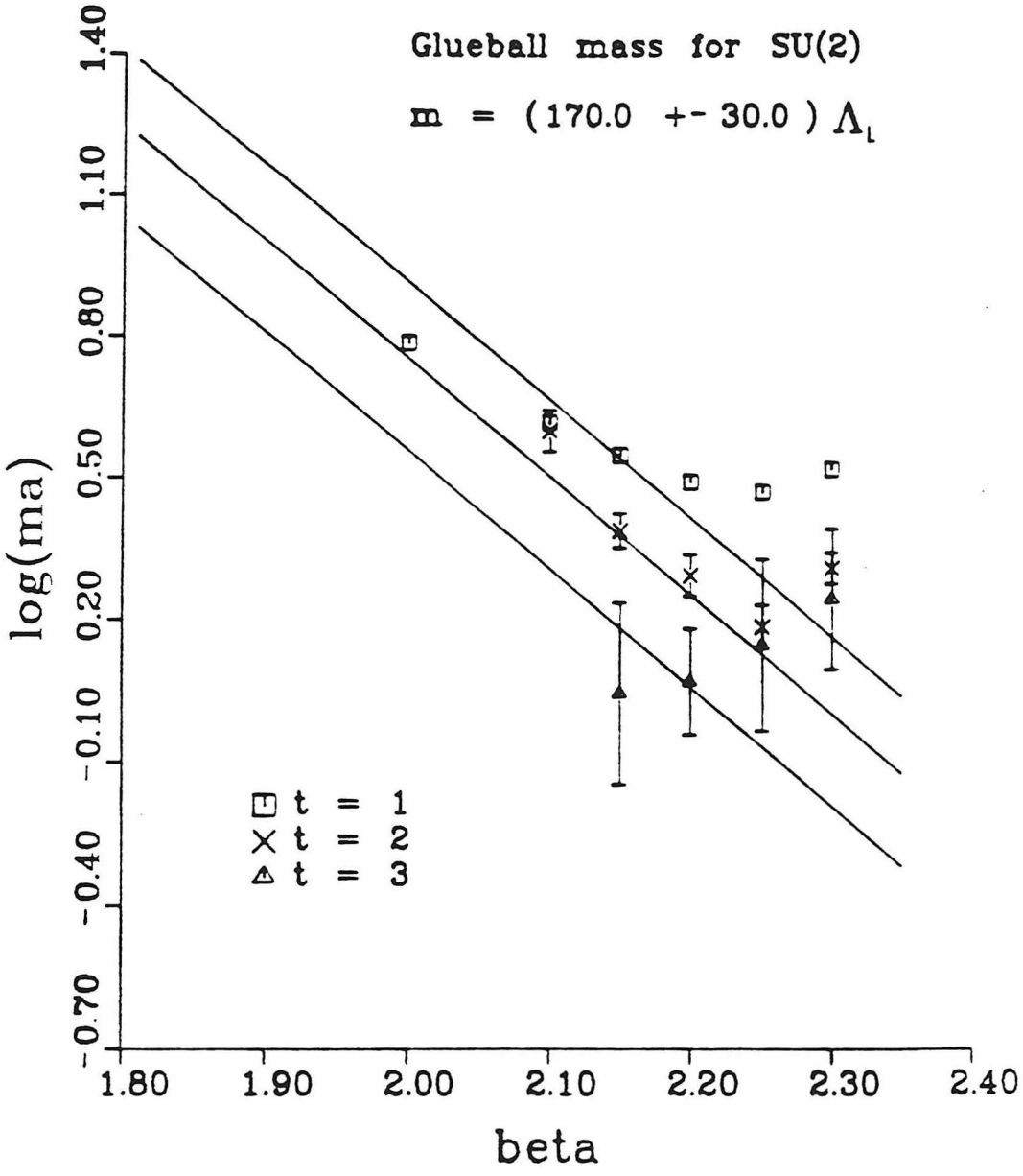

Glueball mass for SU(2)
m = (170.0 +- 30.0) Λ_L
log(ma)
1.40
1.10
0.80
0.50
0.20
-0.10
-0.40
-0.70
1.80
1.90
2.00
2.10
2.20
2.30
2.40
beta
t = 1
t = 2
t = 3

Fig. 2.8

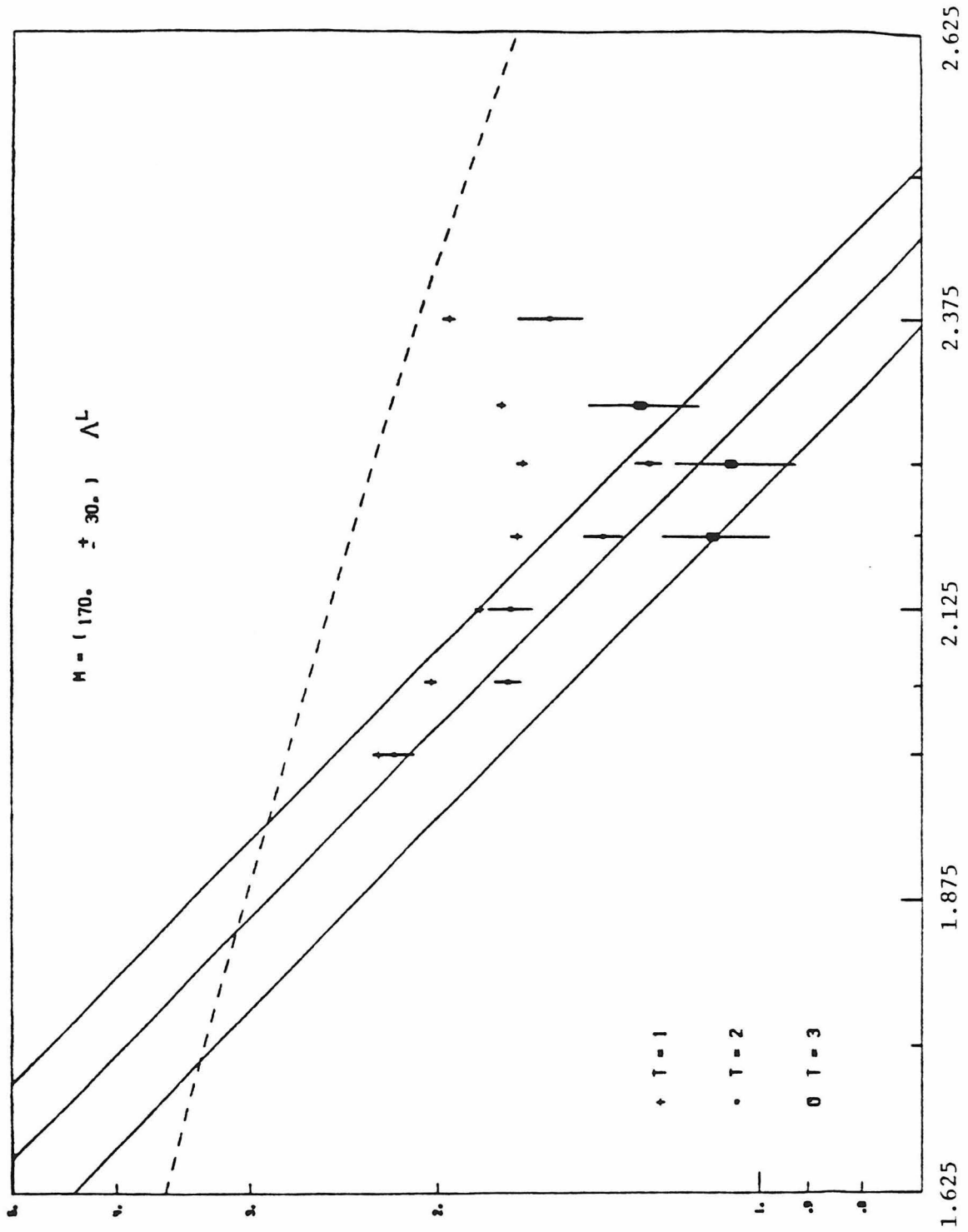


Fig. 2.9

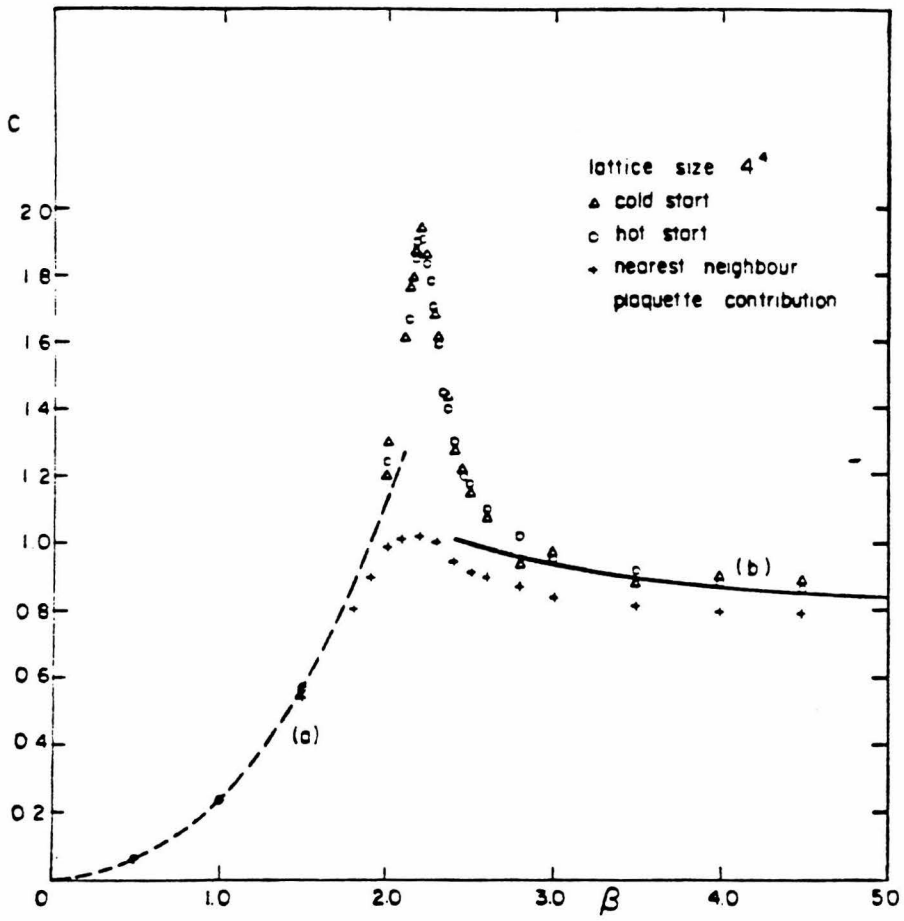


Fig. 2.10

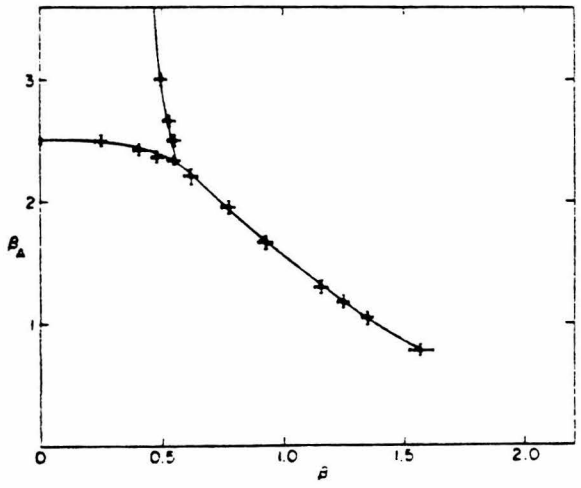


Fig. 2.11

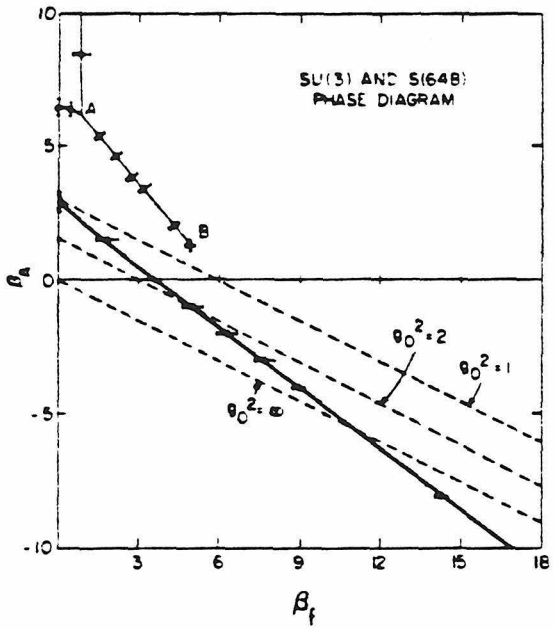

SU(3) AND S(648)
PHASE DIAGRAM
A
B
$\theta_0^2 = 2$
$\theta_0^2 = 1$
$\theta_0^2 = \infty$
β_A
β_f
10
5
0
-5
-10
3
6
9
12
15
18

Fig. 2.12

Chapter III: Migdal-Kadanoff improved actions and the Glueball Mass

Introduction

In the previous chapter we came to the conclusion that, for actions of the purely Wilson form in the range of $2.0 \leq \beta \leq 2.3$, the 0^+ glueball mass does not scale. We attributed this to a nearby phase transition in the extended action plane, (β_F, β_A). One might, therefore, try working in a region in this plane which is farther away from this phase transition, that is, go to β_A negative.

Beyond this simple interpretation, there is another motivation for working with β_A negative. This is the result that, in the Migdal-Kadanoff approximation to the renormalization group on the lattice, one finds that actions starting along the Wilson axis ($\beta_A = 0$), under renormalization, flow towards $\beta_A < 0$ [1]. To the extent that we can trust the Migdal-Kadanoff approximation, this means that theories with $\beta_A < 0$ describe large distance physics much better (i.e. at smaller lattice distances) than theories with $\beta_A = 0$. We work in a coupling region motivated by the Migdal-Kadanoff results, and we do indeed find improved scaling results for the 0^+ glueball. We therefore conclude that this set of actions is superior to the usual Wilson actions in the sense that the true large distance physics is more accurately modeled.

In Section 1 we discuss the Migdal-Kadanoff technique for gauge theories. Section 2 describes the results of Bitar, Gottlieb, and Zachos [1] and discuss what they imply for strong coupling and Monte Carlo calculations. We give our Monte Carlo results in Section 3. Checking the consistency of the results with scaling is somewhat problematical. Using a large N resummation result of Grossman and Samuel [2], we find that our results do scale. *Independent of this*

large N resummation, our data show that the mass of the 0^+ glueball divided by the square root of the string tension is a constant for this set of actions, as it should be if scaling has set in. This is not true of the data along the Wilson axis for $2.0 \leq \beta \leq 2.3$.

1. Migdal-Kadanoff Renormalization for Gauge Theories

The standard references for this are the original papers by Migdal [3] and Kadanoff [4]. We will explain the technique along similar lines to the 2 dimensional Ising model example of Chapter I. We start with a 2 dimensional gauge theory, where link decimation can be done exactly.

Consider two neighboring plaquettes, as in Fig. 3.1. We wish to integrate out the link labeled U. The integral is:

$$\int dU \, e^{\beta tr(U_1 U)} e^{\beta tr(U^\dagger U_2)} \quad , \tag{III.1.1}$$

where $U_1 = U_a^\dagger U_b^\dagger U_c$, $U_2 = U_d U_e U_f^\dagger$. To do this integral, it is convenient to expand in the characters of the group (traces of various irreducible representation matrices). Denote the character of the representation ν by χ_ν . They satisfy the orthogonality property,

$$\int dV \, \chi_\nu(U V) \chi_\mu(V^\dagger W) = \frac{\delta_{\nu\mu}}{d_\nu} \chi_\nu(U W) \quad , \tag{III.1.2}$$

where d_ν is the dimensionality of the representation ν and is equal to $\chi_\nu(1)$.

For SU(2) the characters are quite simple. The "spin" j representation matrix is:

$$U_j = e^{i \vec{J} \cdot \hat{n} \varphi} \quad , \tag{III.1.3}$$

where $\vec{J}$ are the spin j generators, and $\vec{\varphi} = \varphi \hat{n}$ are the 3 parameters of the

rotation. The character of this representation is the trace of this matrix and is given by:

$$\chi_j = \sum_{m=-j}^{j} e^{im\varphi} = \frac{\sin(2j+1)\frac{\varphi}{2}}{\sin\frac{\varphi}{2}} . \qquad \text{(III.1.4)}$$

Expand the exponential of each plaquette in terms of characters:

$$e^{\beta tr(U_p)} = \sum_{\nu} c_\nu(\beta)\, d_\nu \chi_\nu(U_p) . \qquad \text{(III.1.5)}$$

The integral can now be easily done:

$$\int dU\, e^{\beta tr(U_1 U)} e^{\beta tr(U^\dagger U_2)} = \int dV \sum_{\nu\mu} c_\nu(\beta)\, c_\mu(\beta)\, d_\nu\, d_\mu \chi_\nu(U_1 U)\, \chi_\mu(U^\dagger U_2)$$

$$= \sum_{\nu} [c_\nu(\beta)]^2 d_\nu \chi_\nu(U_1 U_2) . \qquad \text{(III.1.6)}$$

This can be continued to the integration of the links in the other direction (the horizontal links in Fig. 3.1). Each plaquette will add another power of $c_\nu(\beta)$. After integrating a, f of Fig. 3.1, we will have,

$$\sum_{\nu} [c_\nu(\beta)]^4 d_\nu \chi_\nu(U_{2\times 2}) , \qquad \text{(III.1.7)}$$

where $U_{2\times 2}$ is the ordered product of matrices around the 2×2 square.

This was for a scale change of 2. For a scale change of λ, the above immediately generalizes to:

$$\sum_{\nu} [c_\nu(\beta)]^{\lambda^2} d_\nu \chi_\nu(U_{\lambda\times\lambda}) . \qquad \text{(III.1.8)}$$

This result is exact for integer λ. As in the Ising model, we assume it is true for all $\lambda \geq 1$. This can, again, be justified by integrating out only some of the links

and then averaging.

In higher dimensions, lines of links cannot be integrated out exactly. We resort to a procedure in which the troublesome plaquettes are shifted to neighboring planes. This is the analog of the bond shifting operation in the Ising model. This means that some plaquettes will have their strength increased by a factor of λ:

$$e^{\beta tr(U_p)} \rightarrow e^{\lambda \beta tr(U_p)} = \left[\sum_\nu c_\nu(\beta)\, d_\nu \chi_\nu(U_p)\right]^\lambda \quad . \tag{III.1.9}$$

As in the Ising case, for λ=2 the isotropy is lost under these transformations. As $\lambda \rightarrow 1$ it is recovered, however, and we write for the full transformation in d dimensions:

$$e^{\beta tr(U_p)} = \sum_\nu c_\nu(\beta)\, d_\nu \chi_\nu(U_p) \rightarrow \left[\sum_\nu [c_\nu(\beta)]^{\lambda^2} d_\nu \chi_\nu(U_p)\right]^{\lambda^{d-2}} \quad . \tag{III.1.10}$$

The entire term is raised to the power λ^{d-2} since, in d dimensions, to integrate out a link requires the shifting of $d-2$ planes of plaquettes.

This defines the Migdal-Kadanoff renormalization transformation. Starting with some arbitrary action, under the application of this transformation traces of plaquette matrices in all irreducible representations of the group will appear. Terms more non-local than the plaquette are not included in this approximation. The systematic improvement of the Migdal-Kadanoff program given by Martinelli and Parisi [5] can, in principle, find the non-local terms also (no one has yet attempted it for gauge theories since it is technically difficult).

1. The Results of Bitar, Gottlieb and Zachos

Bitar, Gottlieb and Zachos have applied the Migdal-Kadanoff transformation to SU(2). They implement Eq. (III.1.10) numerically by expanding their action in terms of the first 20 characters of SU(2). Their major result is that they are able to accurately reproduce the phase diagram of SU(2) in the β_F,β_A plane, which is obtained from Monte Carlo calculations. What interests us, however, is that they also map out the renormalization flows of theories in the β_F,β_A plane. Let us quickly review their results.

After many Migdal-Kadanoff transformations, the coupling between different plaquettes becomes weak (the plaquettes after several transformations are "large" plaquettes - they represent large scale physics). Once this happens, the remaining plaquettes can be integrated out as if they were independent, yielding an expression for the free energy per plaquette. Let us denote the action after N applications of the Migdal-Kadanoff transformation by $S(U_p,\lambda^N a)$, and expand it in characters as:

$$e^{-S(U_p,\lambda^N a)} = \sum_\nu c_\nu(\lambda^N a)\, d_\nu \chi_\nu(U_p) \quad . \tag{III.2.1}$$

Then, integration of a single plaquette gives,

$$\int dU_p\, e^{-S(U_p,\lambda^N a)} = \int dU_p \sum_\nu c_\nu(\lambda^N a)\, d_\nu \chi_\nu(U_p) \tag{III.2.2}$$

$$= c_0(\lambda^N a) \quad ,$$

and we get for the free energy per plaquette ($F = \ln Z$),

$$F = \lim_{N\to\infty} \frac{1}{\lambda^{4N}} \ln c_0(\lambda^N a) \quad . \tag{III.2.3}$$

F is a function of the starting action. Suppose we start somewhere in the β_F,β_A

plane and write the vector, (β_F,β_A) as $\beta(c_F,c_A)$, where $c_F^2 + c_A^2 = 1$. Then, the change of F with β gives us the average action,

$$-<S_p> = \beta \frac{\partial F}{\partial \beta} \quad , \tag{III.2.4}$$

and another derivative gives us the specific heat:

$$C = \beta^2 \frac{\partial^2 F}{\partial \beta^2} \quad . \tag{III.2.5}$$

These derivatives were found by finding F, for a range of β's, via Eq. (III.2.3) and then numerically differentiating.

Phase transitions are signaled by singular behavior in the specific heat, C. By identifying sharp behavior (i.e. cusps, bumps) in C with phase transitions, the phase diagram shown in Fig. 3.2 was produced. Fig. 3.2 also shows the locations of phase transitions as determined from Monte Carlo calculations, and it is found that the two methods agree quite well. Though the Migdal-Kadanoff recursion technique seems to miss the order of the transition (in [1], bumps in C are seen instead of the expected δ function), the locations of the phase transitions are obtained accurately. This is consistent with what has been seen in other models [5,6]. The Migdal-Kadanoff recursion formula, Eq. (III.1.10), has one free parameter, λ, appearing. From the derivation of Eq. (III.1.10), we know that λ should be kept near 1. In [1], λ is fixed by requiring that the location of the phase transition along the axis β_F=0 is obtained correctly. This gives λ=1. Different choices of λ, in the range 1.05 to 1.15, give qualitatively the same phase diagram, but with critical lines moving by approximately 10%.

The Migdal-Kadanoff transformation is able to reproduce the correct phase diagram in the β_F,β_A plane. Beyond this, the actual renormalization flows of various lattice actions can be studied. This is done by starting with some action

in the β_F,β_A plane, converting to the character expansion (Eq. (III.2.1)), renormalizing via Eq. (III.1.10), and then projecting back onto an action containing the first 4 representations: spin $\frac{1}{2}$, 1, $\frac{3}{2}$, and 2. It is found that the spin $\frac{3}{2}$ and 2 coefficients stay much smaller than β_F or β_A. A theory starting in the β_F,β_A plane tends to stay in that plane; theories starting out of the plane tend to flow onto it. The renormalization flows in the plane are shown in Fig. 3.3.

All flows eventually go to the origin, $\beta_A=\beta_F=0$. This is simply due to the fact that as one applies the Migdal-Kadanoff transformation again and again, the plaquettes represent physics on larger and larger length scales, and, therefore, become weakly coupled (β small). The interesting feature of Fig. 3.3, at least for our purposes, is that theories starting on the Wilson axis ($\beta_A=0$) tend to flow downwards ($\beta_A<0$), and coalesce onto a line. The parts of the action which are irrelevant for large scale physics are dying out as the actions flow onto this stable line. For any action in the plane, there is an action on the stable line which has the same large distance (in physical units) behavior, but at smaller lattice distances. Of course, this statement is true only to the extent that the Migdal-Kadanoff technique is accurate.

Suppose the Migdal-Kadanoff approximation was exact, so that the large distance physics of the initial action would be exactly preserved under the transformations. Then we could work on the stable line very close to the point β_F,$\beta_A=0$. Near this point, the lattice correlation length is small. This would mean that strong coupling expansions along this line might converge much more rapidly than they do for the simple $\beta_A=0$ action. Monte Carlo calculations would also become more reliable as the lattice correlation length is made small.

In actual fact, though, we do not trust the Migdal-Kadanoff approximation completely. For this reason, we work on the stable line of the flows coming from this approximation, but at couplings closer to the naive continuum limit, $g \to 0$.

In Chapter II we interpreted the lack of scaling of the glueball mass as being due to a nearby critical point in the β_F,β_A plane. We thereby came to the conclusion that one should work in the $\beta_A<0$ coupling region. This conclusion is now strengthened by the results discussed in this section. The Migdal-Kadanoff transformation tells us that actions along the stable line of Fig. 3.3 are perhaps closer to the continuum limit than those along the $\beta_A=0$ axis. In the next section we will present our Monte Carlo results obtained along the stable line.

2. Results

We have calculated the 0^+ glueball mass in the β_F,β_A plane along the line $\beta_A=-.24\beta_F$, for $2.8\leq\beta_F\leq3.2$. These couplings lie along the stable line of Fig. 3.3. Though our choices of couplings are motivated by the Migdal-Kadanoff renormalization technique, our results do not rely on the accuracy of the technique.

The glueball mass was, again, found by the MCVM method described in Chapter II. The calculation was done on the 4 node parallel processor using a $4^3\times8$ lattice. The set of operators over which the minimization was done is the same as that used previously - the set of 5 loops shown in Fig. 2.6. We did try including an additional operator, the 1×1 loop in the adjoint representation, but we found it made little contribution. After approximately 600 hours of running, we arrived at the results shown in Fig. 3.4.

For the results at $\beta_F=2.8,2.9$, the glueball seems to be well isolated at the second time slice, $t=2$. For $\beta_F=3.0,3.2$, at least the $t=3$ time slice seems to be necessary to isolate the state. In contrast to the situation on the $\beta_A=0$ axis, the lower envelope of the masses seems to be falling smoothly and monotonically. There is no sign of a "bump" in the masses as there is on the $\beta_A=0$ axis. As expected, the influence of the critical endpoint has decreased.

No obvious signs of lattice artifacts are seen but, the question is, do our masses for this coupling range scale according to the continuum renormalization group? The answer to this is problematical. By examining the coefficient of $F_{\mu\nu}F^{\mu\nu}$ in the naive continuum limit of an action containing both fundamental and adjoint traces, one finds the following relation between g and the β's:

$$\frac{1}{g^2} = \frac{1}{4}\beta_F + \frac{2}{3}\beta_A \quad . \tag{III.3.1}$$

Negative β_A implies larger values for g, and for our range of couplings along the stable line, g^2 ranges from 3.47 to 4.23 (to compared with 1.74 to 2.0 on the Wilson axis). Such large values for g^2 makes one suspicious of any comparison with 2 loop perturbation theory, and this suspicion is, in fact, well founded. For actions containing both adjoint and fundamental representations, the connection between the lattice spacing, the couplings, and a physical mass scale is given by [7] (for SU(2)):

$$\Lambda_L = \frac{1}{a}\left[\frac{6\pi^2}{11}(\beta_F+2\beta_A)\right]^{\frac{51}{121}} \exp\left[-\frac{3\pi^2}{11}(\beta_F+2\beta_A) + \frac{15\pi^2}{22}\frac{\beta_A}{\beta_F+2\beta_A}\right] \quad . \tag{III.3.2}$$

If we convert our mass results to physical units using the above, totally nonsensical results are found. This was also found by Bhanot and Dashen [7]. They computed string tensions in the β_F,β_A plane and found large disagreements with scaling as β_A was taken negative. Perturbation theory has broken down in the regions of couplings in which we work. This shouldn't be too surprising; the whole point of the $\beta_A<0$ region is that large scale physics is accurately modeled at short lattice distances. The only way to achieve this is if non-trivial interactions are included (the higher representations), with the couplings not necessarily changing according to weak coupling perturbation theory.

Perturbation theory can't tell us the relation between our lattice spacing and a physical scale. One way out of this dilemma would be to know exactly which action on the β_A=0 axis corresponded to which set of actions along the stable line. One could start with an action far into the weak coupling regime (g^2 small) on the axis, and then scale it by a factor, λ, many times, eventually landing on the stable line. Assuming the Migdal-Kadanoff approximation to be accurate, we could then relate the lattice spacing of the theory on the stable line to the spacing of the initial theory by a factor, λ^N, where N is the number of transformations between the two theories. We would then have our lattice spacing related to a spacing far into the weak coupling regime, where the usual 2 loop formula (Eq. (II.3.5)) would hold. This procedure is unreliable due to the large extrapolation required - the power N is large and we would need to rely on the accuracy of the Migdal-Kadanoff technique for a very large change in length scales.

A comparison with continuum perturbation theory would be possible if the expression were known to greater than 2 loops. Such a result is, perhaps, available. Grossman and Samuel [2] give an argument based on a large N expansion (SU(N) for N large) which, in effect, resums perturbation theory. They achieve good results in the sense that they are able to agree with string tension data for negative β_A, for which the standard 2 loop result rapidly breaks down. Their lines of constant string tension are shown in Fig. 3.5a. These lines rely on the value of the average plaquette, for which Grossman and Samuel use a weak coupling approximation. We improve, slightly, on their results by using the value for the plaquette as obtained from accurate Monte Carlo calculations [8]. Our improvement is shown in Fig. 3.5b. It is seen that the agreement with the string tensions is improved.

The resummation relates theories with negative β_A to theories with β_A=0, at a new, effective value for β_F . The relation is:

$$\beta_{eff} = \beta_F + \beta_A < tr\, U_p > + \frac{4}{3}\beta_A \left(\frac{1}{2} < tr\, U_p > - 1\right) \quad , \qquad \text{(III.3.3)}$$

where U_p is the plaquette matrix. We can now map our glueball masses onto the Wilson axis using the above result. This is shown in Fig. 3.6. We are quite encouraged by these results. First of all, the lower envelope of our data does seem to be falling with approximately the correct slope. There is no large disagreement, and we make the claim that the data are consistent with scaling. Secondly, the coupling region in which we worked corresponds to, on the Wilson axis, a region of weaker couplings than we were able to directly do on this axis. Therefore, our hope that the Migdal-Kadanoff improved action would be closer to the continuum limit seems to be confirmed. Lastly, our data seem to smoothly join onto the data obtained on the Wilson axis. On this axis at β=2.3, we previously concluded that the third time slice, t=3, was not far enough to isolate the glueball. By pushing to t=4, it is plausible that the mass at this value would agree with the data coming from the β_A<0 region.

One can argue with the reliability of the resummation of Grossman and Samuel. After all, it relies on a large N argument, and N is 2! Our only reason for believing it has any relevance to the N=2 case lies in the fact that it does agree with string tension data quite well.

We can, however, divorce our results from this possibly unreliable resummation by just comparing our masses to the string tension directly. Different physical observables will be affected by the discrete lattice lattice approximation in different ways. If dimensionless ratios are taken, a sign that the calculation has "forgotten" the finite lattice spacing is when such ratios become constant.

The square root of the string tension depends on the lattice spacing as $\frac{1}{a}$, as does the glueball mass. We can regard the large N resummation result as nothing more than a convenient way of fitting the Monte Carlo string tensions to a simple functional form (linear). Our result that the masses, when mapped onto the Wilson axis, follow the the scaling curve can now be reinterpreted as showing that, in the region of couplings in which we have worked, we find the ratio of the glueball mass to the square root of the string tension to be a constant. Fig. 3.7 explicitly shows the ratio of the mass to the "fit" of the string tensions. This result is independent of the, perhaps dubious, large N resummation.

Before ending this section, we would like to point out that our masses change, from β_F=2.9 to β_F=3.2, by a factor consistent with the scale change of the Migdal-Kadanoff transformation. The point (3.2, -.768) is transformed to the point (2.9,-.696) by approximately 2.5 transformations, each of scale change, λ=1.1. This gives a total change in scale of $1.1^{2.5} \approx 1.27$. Our masses change from approximately .80 to 1.05 over this same range, giving a scale change of $\approx$1.31. The two values are equal within our limited statistics, and we believe this shows that the Migdal-Kadanoff approximation is accurate for our region of couplings.

Conclusions

We have shown that actions incorporating a negative component of the adjoint representation are closer to the continuum limit than actions along the Wilson axis. Working on the stable line of the Migdal-Kadanoff approximate renormalization flows, we have shown that the mass of the 0^+ glueball divided by the square root of the string tension is a constant, something which is <u>not</u> true on the Wilson axis for the coupling range which has been studied.

Further work needs to be done in the $\beta_A<0$ region. An obvious extension is to calculate higher states of the glueball spectrum. We did not do this since one expects that scaling is harder to achieve for these states. The wavefunctionals of the higher states must have nodes and hence vary more rapidly in space than the 0^+ . This makes them more sensitive to the discrete lattice. This argument is consistent with the results of [9], which finds that the 2^+ state does not scale. If the $\beta_A<0$ actions are modeling continuum physics more accurately, however, we expect to see scaling for these higher states sooner than on the Wilson axis.

Improvements on the Migdal-Kadanoff scheme are also needed. The method of Martinelli and Parisi [5] is a possibility, though the method of renormalizing via Monte Carlo (the "Monte Carlo Renormalization Group") proposed by Wilson [10], is perhaps the most powerful method. With such improvements in the lattice theories, coupled with more powerful computing capabilities, it is reasonable to expect that reliable calculations of the glueball spectrum for QCD are not far away.

References

[1] K. Bitar, S. Gottlieb, C. Zachos, Phys. Rev. D26, 2853 (1982).

[2] B. Grossman, S. Samuel, Phys. Lett. 120B, 383 (1983).

[3] A. Migdal, Sov. Phys.-JETP 42, 413 (1975)

[4] L. Kadanoff, Ann. Phys. (N.Y.) 100, 359 (1976).

[5] G. Martinelli, G. Parisi, Nucl. Phys. B180[FS2], 201 (1981).

[6] G. Martinelli, P. Menotti, Nucl. Phys. B180[FS2], 483 (1981).

[7] G. Bhanot, R. Dashen, Phys. Lett. 113B, 299 (1982).

[8] B. Berg, J. Stehr, Z. Phys. C9, 349 (1981).

[9] B. Berg, A. Billoire, DESY preprint, DESY 82-079 (1982).

[10] K. G. Wilson, Cargese lecture notes (1979).

Figure Captions

[3.1]Two neighboring plaquettes. We integrate out the link marked as U.

[3.2]The phase diagram of reference [1]. The points are the locations of phase transitions as determined from the Migdal-Kadanoff recursion; the dashed lines are the Monte Carlo results. The dotted line is the extrapolation of the critical line to the Wilson axis and intersects it at the location of the peak in the specific heat (S). The normalizations of the couplings are different between our work and that of [1]; denoting their couplings by $\beta^{[1]}$, the relation is:

$$\beta_F^{US} = 2\beta_F^{[1]} \quad , \quad \beta_A^{US} = 3\beta_A^{[1]} \quad .$$

[3.3]The renormalization flows of reference [1].

[3.4]Our data in the negative β_A region. The horizontal axis is β_F .

[3.5](a) The Grossman, Samuel [2] prediction for the lines of constant string tension. The points are from Monte Carlo [7] and the cross-hatched lines are the prediction. (b) Our improvement, using the Monte Carlo value of $< tr\, U_p >$ instead of the weak coupling result.

[3.6]Our results for the glueball mass mapped onto the Wilson axis. The data on the left hand side ($2.0 \leq \beta \leq 2.25$) are the old data coming from runs on the axis; the data toward the right($2.28 \leq \beta \leq 2.41$) are the new data coming from the $\beta_A < 0$ region.

[3.7]The glueball mass divided by the square root of the string tension. (a) For the $\beta_A < 0$ region, (b) the $\beta_A = 0$ region.

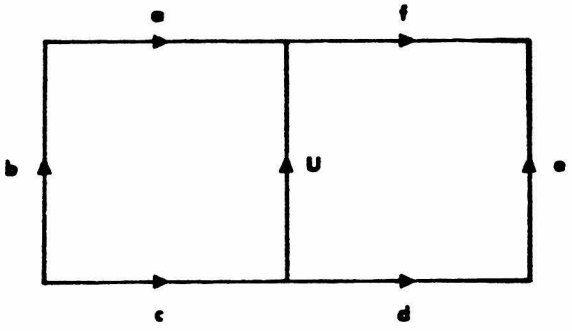


Fig. 3.1

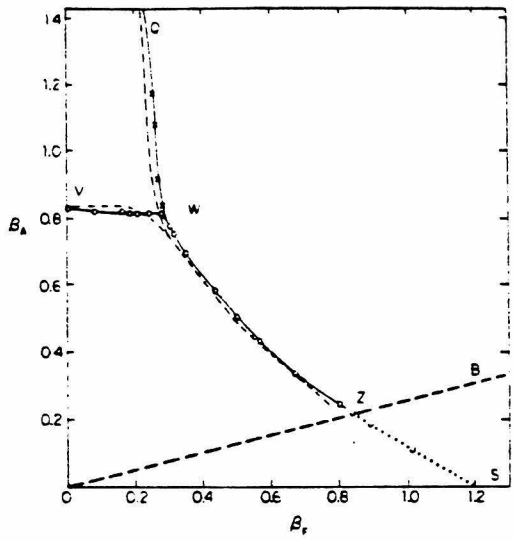


Fig.3.2

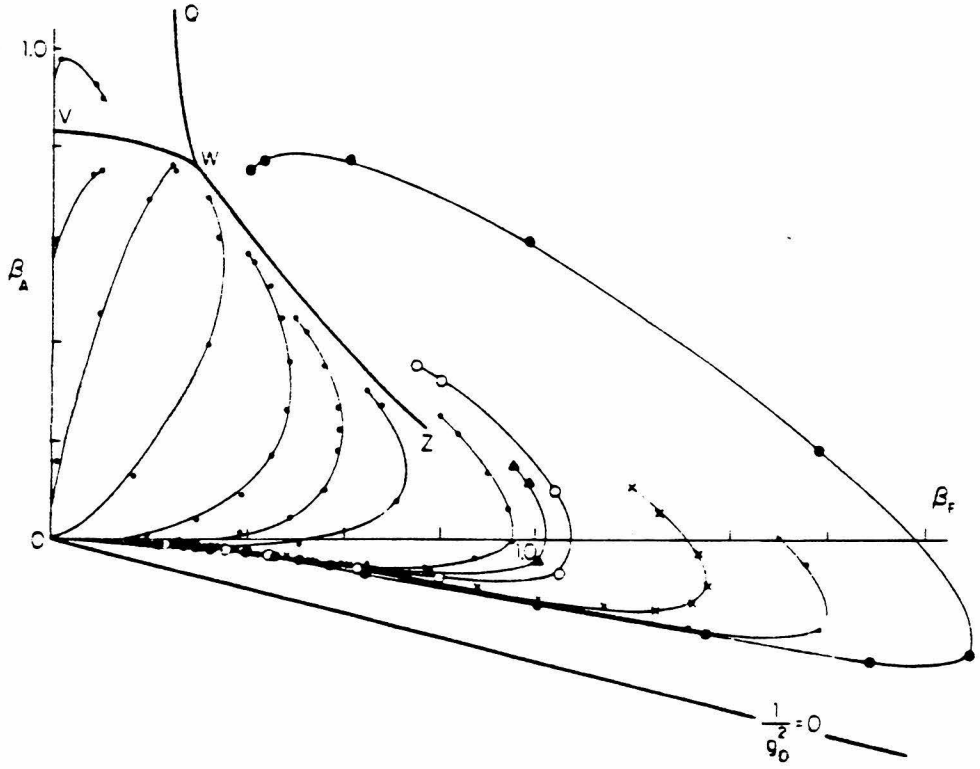


Fig. 3.3

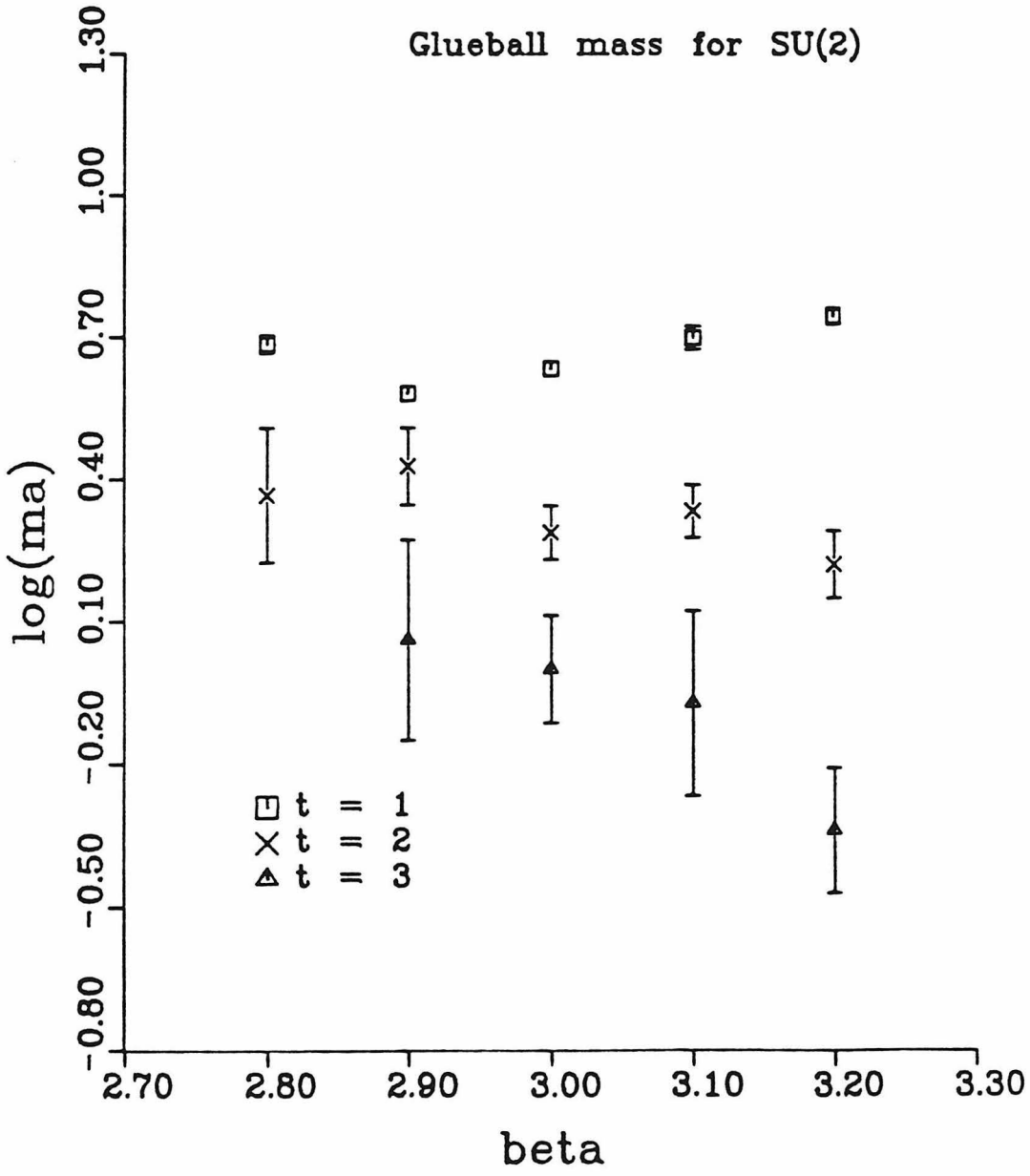

Glueball mass for SU(2)
log(ma)
1.30
1.00
0.70
0.40
0.10
-0.20
-0.50
-0.80
t = 1
t = 2
t = 3
2.70
2.80
2.90
3.00
3.10
3.20
3.30
beta

Fig. 3.4

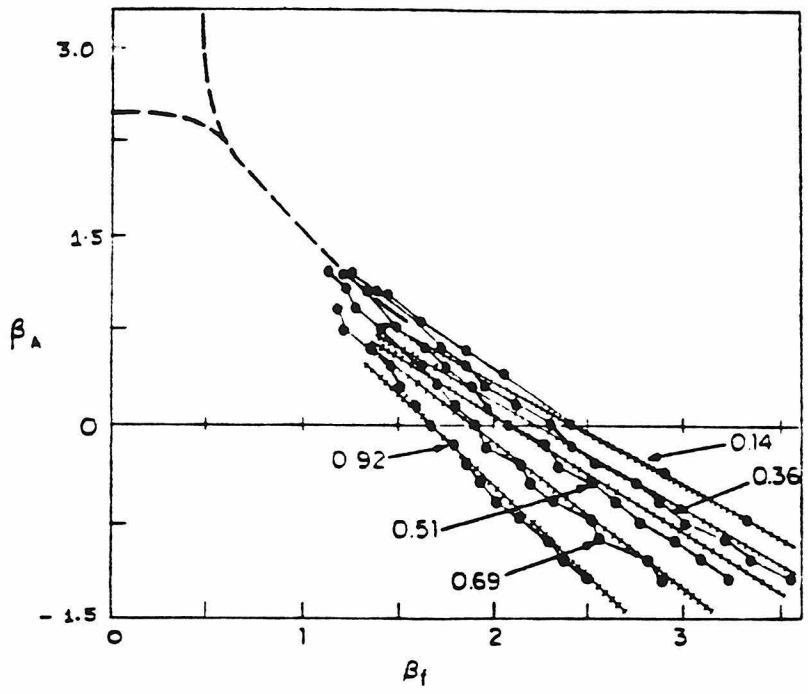


Fig. 3.5a

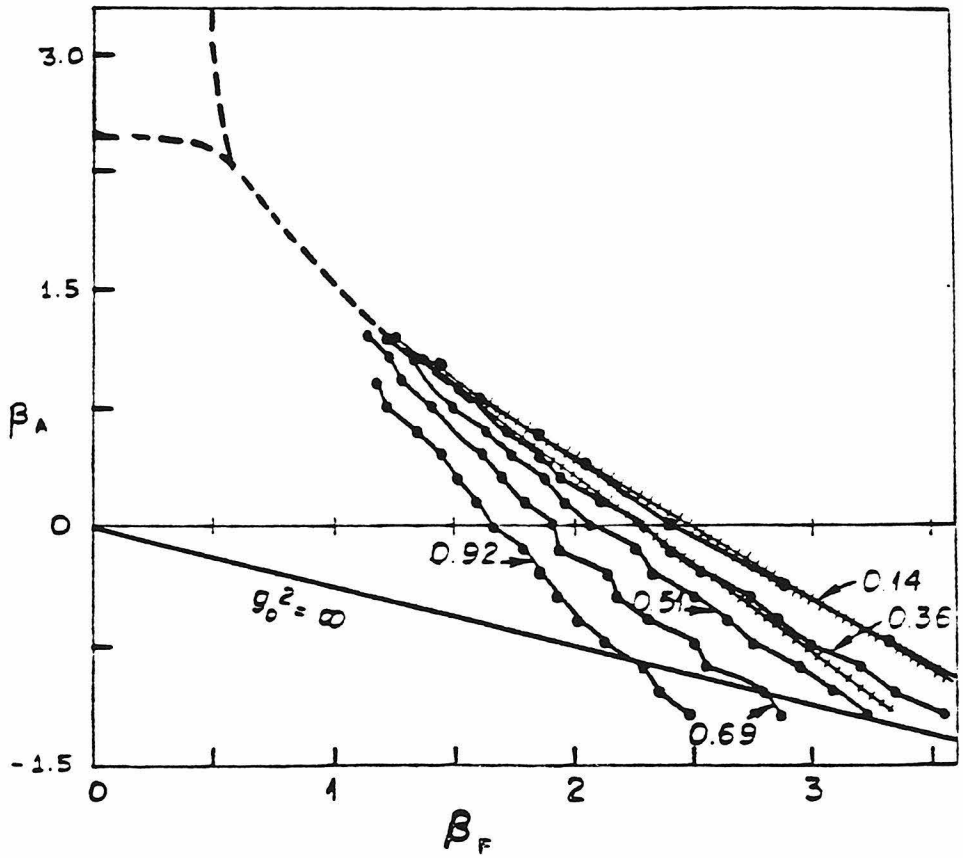


Fig. 3.5b

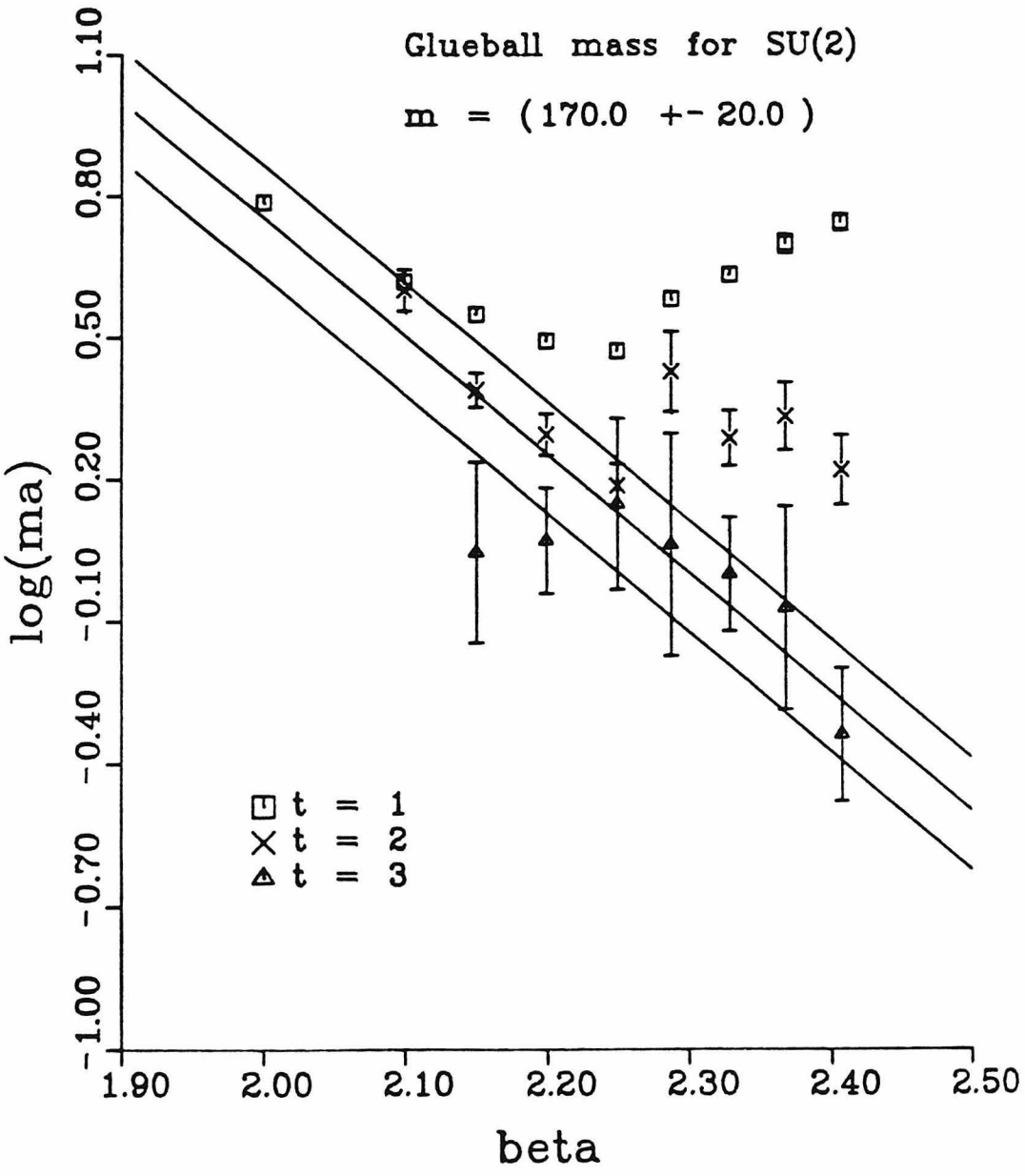
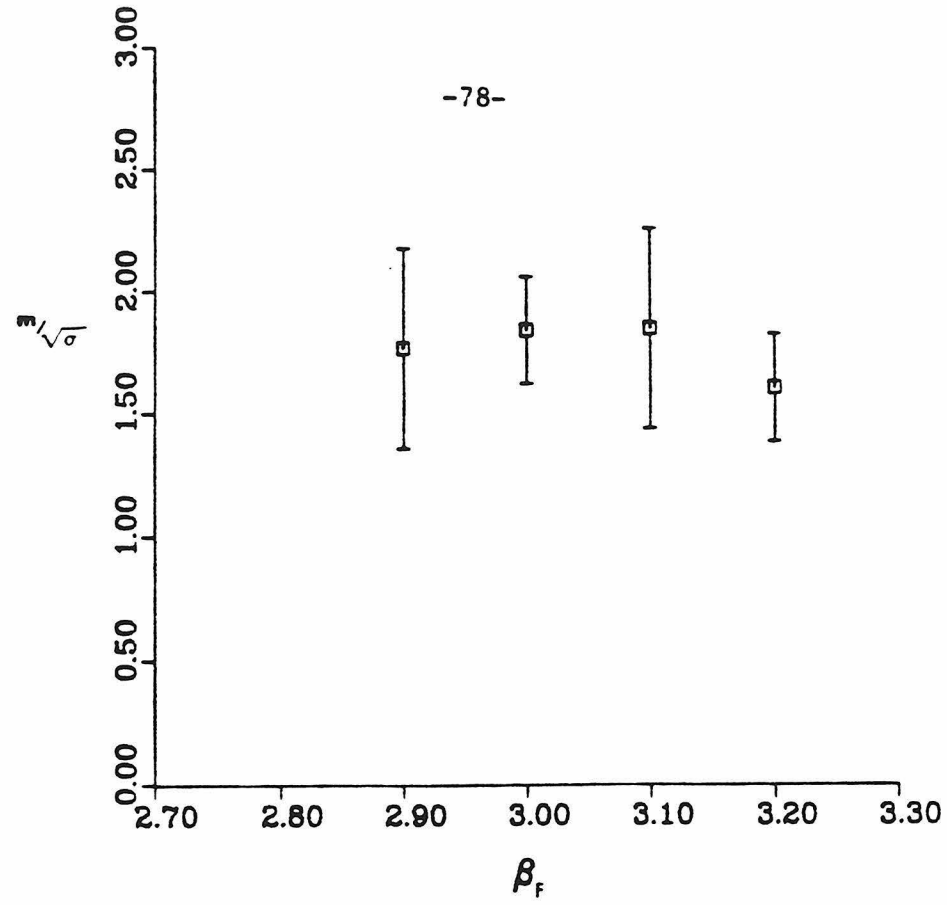
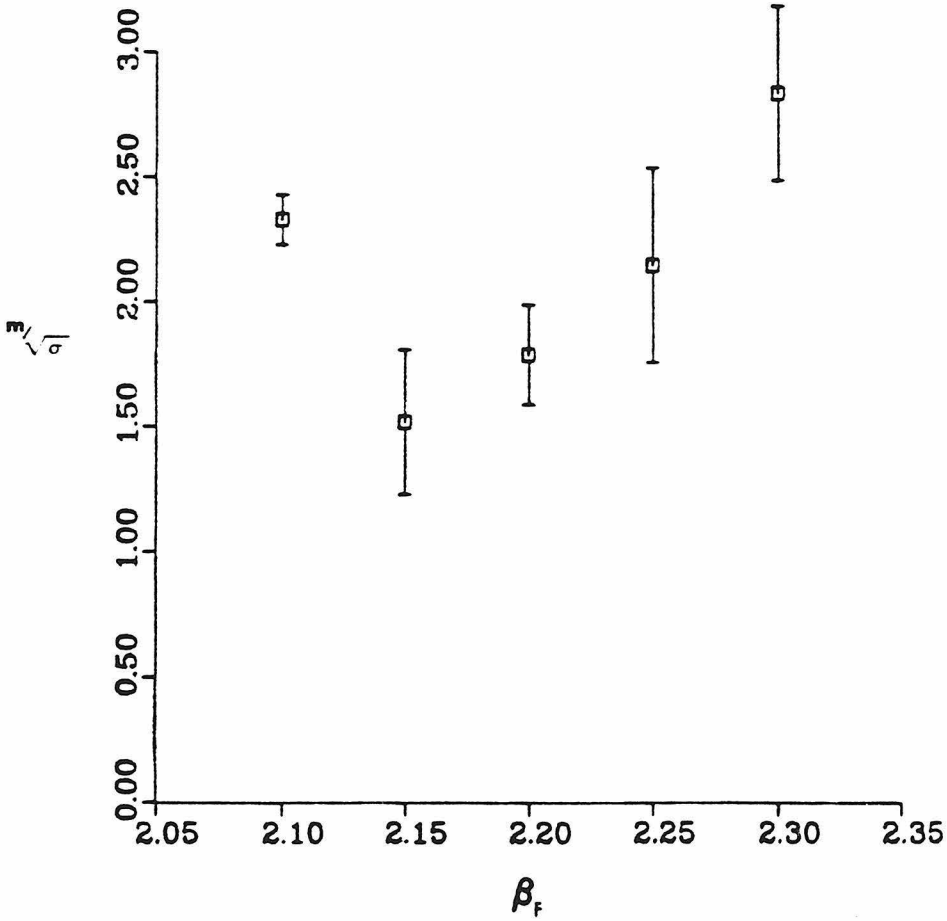

Glueball mass for SU(2)
m = (170.0 +- 20.0)
log(ma)
1.10
0.80
0.50
0.20
-0.10
-0.40
-0.70
-1.00
1.90
2.00
2.10
2.20
2.30
2.40
2.50
beta
t = 1
t = 2
t = 3

Fig. 3.6


3.00
2.50
2.00
1.50
1.00
0.50
0.00
2.70
2.80
2.90
3.00
3.10
3.20
3.30
β_F

(a)


3.00
2.50
2.00
1.50
1.00
0.50
0.00
2.05
2.10
2.15
2.20
2.25
2.30
2.35
β_F

(b)

Fig. 3.7

Chapter IV: Numerical Fermion Techniques

Introduction

The inclusion of dynamical fermions in Monte Carlo calculations is one of the most important problems in the field of lattice gauge theories. Even if one is optimistic and believes that the "quenched" approximation is accurate, it is known that it cannot reproduce all features of the hadronic mass spectrum. An example is the mixing of glueballs with flavor singlet, $\bar{q}q$ states and the η,η' splitting. Secondly, the calculations cannot be considered to be based on first principles until dynamical fermions have been included.

At the present time, there seem to be two promising methods for including dynamical fermions: the stochastic technique of Kuti [1], and the pseudo fermion method invented by Fucito, Marinari, Parisi, and Rebbi [2]. These two methods have a "chance" in the sense that they can accurately represent the non-local interaction induced by the dynamical fermions, and they can run in reasonable computation times. This last condition is necessary so that adequate lattice sizes and statistics can be gathered to calculate interesting observables.

In this chapter, we will study the pseudo fermion technique by comparing Monte Carlo results obtained with this method to those obtained from an exact algorithm based on Gauss-Seidel inversion. In Section 1, we will explain the exact Gauss-Seidel algorithm and also briefly mention the stochastic technique. The pseudo fermion method is introduced, and results of the application to 1+1 QED are given, in Section 2. We show that, in a coupling regime where the dynamical fermions have a nontrivial effect, the method seems to work well. Section 3 gives simple arguments showing why the method works better than

one might naively expect. Section 4 moves on to study the real theory, 4 dimensional SU(3) with dynamical fermions, on small lattices. Comparing with the exact (Gauss-Seidel) algorithm, we again find encouraging agreement with the pseudo fermion technique. We also give some evidence that any systematic bias (associated with the "breaking" of the Markov process which generates the field configurations) is small. The results of the pseudo fermion method applied to the Schwinger model has been previously described in [3].

1. **Exact Algorithm, Stochastic Method**

The basic formalism for fermions on the lattice was already introduced in Chapter I. In the following, we will use the Wilson formulation for fermions on the lattice. Considering the Metropolis method for updating the gauge field configuration, we found that the moving of a gauge link,

$$U \to U + \delta U \,, \tag{IV.1.1}$$

induced a change in the matrix Q,

$$Q \to Q + \delta Q \,, \tag{IV.1.2}$$

and the required ratio of determinants is:

$$\frac{\det(\, Q + \delta Q \,)}{\det(\, Q \,)} = \det(\, 1 + Q^{-1}\, \delta Q \,) \,. \tag{IV.1.3}$$

Q is here repeated for completeness,

$$Q_{x\alpha A, y\beta B} = \delta_{xy}\, \delta_{\alpha\beta}\, \delta_{AB} - \kappa\, M_{x\alpha A, y\beta B} \quad , \tag{IV.1.4}$$

with,

$$M_{x\alpha A, y\beta B} = (1+\gamma^{\mu})_{\alpha\beta}\, (U_{x,\hat{\mu}})_{AB}\, \delta_{x, y-\hat{\mu}} + (1-\gamma^{\mu})_{\alpha\beta}\, (U^{\dagger}_{x-\hat{\mu},\hat{\mu}})_{AB}\, \delta_{x, y+\hat{\mu}} \,. \tag{IV.1.5}$$

x,y are lattice sites, α,β are Dirac indices, and A,B are color indices.

If a single link is changed, δQ will have only two non-zero entries (neglecting spin and color) and the determinant of Eq. (IV.1.3) reduces to the determinant of a small matrix, requiring only a few elements of Q^{-1} [4]. To be explicit, suppose we change $U_{x,\hat{\mu}}$ for some fixed x, $\hat{\mu}$

$$U_{x,\hat{\mu}} \rightarrow U_{x,\hat{\mu}} + \delta U_{x,\hat{\mu}} . \tag{IV.1.6}$$

Then,

$$\delta Q_{w\sigma,y\beta} = -\kappa(1+\gamma^{\mu})_{\sigma\beta}\,\delta U_{x,\hat{\mu}}\,\delta_{w,x}\,\delta_{y,x+\hat{\mu}} - \kappa(1-\gamma^{\mu})_{\sigma\beta}\,\delta U^{\dagger}_{x,\hat{\mu}}\,\delta_{w,x+\hat{\mu}}\,\delta_{y,x} , \tag{IV.1.7}$$

where we have suppressed color indices. Thus, the matrix $Q^{-1}_{z\rho,w\sigma}\,\delta Q_{w\sigma,y\beta}$ is non-zero only when $y=x$ or $y=x+\hat{\mu}$. This implies,

$$\det(1+Q^{-1}\delta Q) = \det\begin{bmatrix} 1+Q^{-1}_{x\alpha,x+\hat{\mu}\rho}\,\delta Q_{x+\hat{\mu}\rho,x\beta} & Q^{-1}_{x\alpha,x\rho}\,\delta Q_{x\rho,x+\hat{\mu}\beta} \\ Q^{-1}_{x+\hat{\mu}\alpha,x+\hat{\mu}\rho}\,\delta Q_{x+\hat{\mu}\rho,x\beta} & 1+Q^{-1}_{x+\hat{\mu}\alpha,x\rho}\,\delta Q_{x\rho,x+\hat{\mu}\beta} \end{bmatrix} . \tag{IV.1.8}$$

In 4 dimensions, taking the Dirac indices into account, this is an 8×8 matrix (ignoring the color indices). We see that, by finding a few elements of Q^{-1}, it is possible to evaluate the ratio of fermion determinants exactly. These elements can be found (with very high accuracy) via a Gauss-Seidel iteration [5] and this forms the basis for the "exact" algorithm with which we will compare our pseudo fermion results. The method of Gauss-Seidel iteration is explained in Appendix 2. Though it is a relatively slow way to invert Q, it converges geometrically and so gives a very accurate estimate of Q^{-1}. The exact algorithm proceeds as follows. A particular link is moved, inducing Q to change by δQ. Gauss-Seidels are then run to find the elements of Q^{-1} needed for the ratio of determinants. Once this ratio is found, it is combined with the change in the pure gauge action, ΔS_{gauge}, and it is decided whether or not to accept the link move, δU. The

entire process is then repeated to update the next link.

This method is very slow. Entire Gauss-Seidel iterations must be run to update a single link. Since the time for each Gauss-Seidel grows as the volume of the lattice, the total time for the exact algorithm grows as the square of the lattice volume. This method is not really compatible with present day Monte Carlos, which have very limited statistics. What is the use of evaluating the fermion determinant to .01% accuracy if only 10 sweeps can be run, generating 100% statistical errors? What is needed is a method which can quickly give a rough, 10-20% estimate of the determinant ratio.

The stochastic technique [1] finds Q^{-1} using the expansion,

$$Q^{-1} = \frac{1}{1-\kappa M} = 1 + \kappa M + \kappa^2 M^2 + \cdots \quad . \qquad \text{(IV.1.9)}$$

The structure of M is such that it couples nearest neighbor sites. Therefore, the term $\kappa^L (M^L)_{xy}$ consists of the sum of all random walks of length L which start at x and end at y. For each such path, the product $\prod (1+\gamma^\mu) U$ is evaluated; these are added for all the paths to give M^L . The stochastic algorithm samples the series of Eq. (IV.1.9) by constructing such random walks, with the probability that any particular walk will be constructed proportional to κ^L .

A problem with this method is the huge number of random paths in 4 dimensions. If x and y are nearest neighbors (needed for the determinant ratio), the number of paths between them of length 5 is 114, of length 7, 3618, and of length 9, 122100 ! It is important to realize that a length 9 path is not that long; the farthest it can get from x,y is 4 and the distance for the average path is $\sim$ 2. Asymptotically, the number of paths of length L grows as 7^L [6]. The crucial question is: for a given correlation length, ξ, what fraction of all possible paths of length L needs to be sampled before a reasonably accurate estimate of Q^{-1} is found?

2. The Pseudo Fermion Method

The pseudo fermion technique allows one to find many elements of Q^{-1} simultaneously so that many links can be updated, gaining a large factor in speed. This technique relies on the result,

$$Q^{-1}_{ji} = \frac{\int [d\bar{\varphi}][d\varphi]\bar{\varphi}_i \varphi_j \, exp[-\sum_{lm}\bar{\varphi}_l Q_{lm}\varphi_m]}{\int [d\bar{\varphi}][d\varphi] \, exp[-\sum_{lm}\bar{\varphi}_l Q_{lm}\varphi_m]} , \qquad \text{(IV.2.1)}$$

where the φ's carry Dirac and color indices, but are bosonic (commuting) variables instead of Grassmann. This is numerically well defined only if Q is hermitian so that all it's eigenvalues are real. Since our Q is not, we cannot use Eq. (IV.2.1) as it stands. Using $\det Q>0$ and $\det Q^\dagger=\det Q$,[1] one can square Q to get a hermitian matrix:

$$\det Q = \sqrt{\det Q \det Q} = \sqrt{\det Q^\dagger Q} \qquad \text{(IV.2.2)}$$

and

$$\frac{\det(Q+\delta Q)}{\det Q} = \left[\frac{\det(Q^\dagger Q+\delta(Q^\dagger Q))}{\det(Q^\dagger Q)}\right]^{1/2} = [\det(1+(Q^\dagger Q)^{-1}\delta(Q^\dagger Q))]^{1/2} . \qquad \text{(IV.2.3)}$$

Now use Eq. (IV.2.1) for $Q^\dagger Q$:

$$(Q^\dagger Q)^{-1}_{ji} = \frac{\int [d\bar{\varphi}][d\varphi]\bar{\varphi}_i \varphi_j \, exp[-\sum_{lm}\bar{\varphi}_l (Q^\dagger Q)_{lm}\varphi_m]}{\int [d\bar{\varphi}][d\varphi] \, exp[-\sum_{lm}\bar{\varphi}_l (Q^\dagger Q)_{lm}\varphi_m]} . \qquad \text{(IV.2.4)}$$

First, consider the use of this result for the update of a single link. To find the needed elements of $(Q^\dagger Q)^{-1}$, a Monte Carlo is run in the complex field φ, which interacts via the next-nearest neighbor action $\bar{\varphi}Q^\dagger Q\varphi$. As this auxiliary

1. These results can be proved using: $\gamma_5 Q \gamma_5 = Q^\dagger$.

Monte Carlo runs, the correlations $\bar{\varphi}_i \varphi_j$ are measured, providing an estimate for $(Q^\dagger Q)^{-1}_{ji}$ which converges statistically (as $\frac{1}{\sqrt{N}}$, where N is the number of sweeps in the pseudo variables) to the correct value. These estimates are then used in Eq. (IV.2.3) to calculate the ratio of determinants. The analog of Eq. (IV.1.8) involves a much larger matrix due to the next-nearest neighbor structure of $Q^\dagger Q$. Once the present link is updated, a new pseudo Monte Carlo is run to update the next link. In the limit of an infinite number of pseudo-sweeps, $N \to \infty$, this is an exact algorithm, though a very slow one. The improvements in speed come with some approximations.

The idea is to partially break the Markov chain of configuration i evolving out of configuration $i-1$, and so on, and, instead, update several gauge links using the estimates for $(Q^\dagger Q)^{-1}$ from the same pseudo Monte Carlo. The Markov chain is broken since the $(Q^\dagger Q)^{-1}$ estimates will not take into account the links that have been moved since the pseudo Monte Carlo was run - they will be "old" estimates. If P gauge links are updated from the same pseudo Monte Carlo, one gains a factor of P in speed. The idea certainly makes sense for a sparse set of links - the effect on $(Q^\dagger Q)^{-1}$ of the movement of a far away link (distance $>\xi$) is small.

Since δQ can be made small by making the Metropolis hit, δU, small, we can expand the determinant ratio in powers of δU:

$$\det(1+(Q^\dagger Q)^{-1}\delta(Q^\dagger Q)) = 1 + tr\left[(Q^\dagger Q)^{-1}\delta(Q^\dagger Q)\right] + O(\delta U^2) \quad . \qquad \text{(IV.2.5)}$$

The trace is much easier to evaluate than the determinant, but the real improvement in speed comes when this linearizing approximation is applied to an entire sweep through the gauge links. Suppose we sweep through all the gauge links of the lattice, recalculating $(Q^\dagger Q)^{-1}$ for each update (as one rigorously should). Effectively, one is then calculating the product, (denote $Q^\dagger Q$

by W),

$$\frac{\det(W+\delta W_1)}{\det(W)}\,\frac{\det(W+\delta W_1+\delta W_2)}{\det(W+\delta W_1)}\cdots\frac{\det(W+\delta W_1+\cdots+\delta W_n)}{\det(W+\delta W_1+\cdots+\delta W_{n-1})}. \quad \text{(IV.2.6)}$$

This is, to linear order in δU,

$$(1+tr W^{-1}\delta W_1)(1+tr(W+\delta W_1)^{-1}\delta W_2)\cdots(1+tr(W+\delta W_1+\cdots+\delta W_{n-1})^{-1}\delta W_n)$$

$$=(1+tr W^{-1}\delta W_1)(1+tr W^{-1}\delta W_2-tr W^{-2}\delta W_1\delta W_2+\cdots)\cdots(1+tr W^{-1}\delta W_n+O(\delta U^2))$$

$$=(1+tr W^{-1}\delta W_1)(1+tr W^{-1}\delta W_2)\cdots(1+tr W^{-1}\delta W_n)+O(\delta U^2) \quad . \quad \text{(IV.2.7)}$$

We see that, if one is linearizing the determinant, it is formally just as good an approximation to use the same $(Q^\dagger Q)^{-1}$ for all of the link updates. That is, in this approximation, *the pseudo Monte Carlo needs to be run only once in order to update the entire gauge field configuration.*

Of course, this is only formally true - one could just as well run the pseudo Monte Carlo at the beginning of a run and then use it to update the gauge configuration 100 times! The errors, though of order $(\delta U)^2$, build up. We will call this error, which is associated with the breaking of the Markov chain, the "systematic" error (it doesn't go away as $N\to\infty$).

Our original intent was to study the severity of the systematic error by updating subsets of all the gauge links from the same pseudo Monte Carlo. As a first step, we did the fastest algorithm, and the one with the largest systematic error - we updated <u>all</u> the gauge links using a single pseudo Monte Carlo. Surprisingly, this seemed to work quite well. Let us move on to some of the numerical results.

Results

We chose, as a testing ground for the pseudo fermion method, the Schwinger Model, QED in 1+1 dimensions. The lattice action is,

$$S[U,\psi,\bar{\psi}] = \beta \sum_{x\mu\nu} [1 - \cos\vartheta_{\mu\nu}(x)] + \sum_{ij} \bar{\psi}_i Q_{ij} \psi_j \quad . \tag{IV.2.8}$$

where $\beta(=\frac{1}{g^2})$ is the coupling constant and $\vartheta_{\mu\nu}$ is given by,

$$\vartheta_{\mu\nu} = \vartheta_\mu(x) + \vartheta_\nu(x+\hat{\mu}) - \vartheta_\mu(x+\hat{\nu}) - \vartheta_\nu(x) \quad . \tag{IV.2.9}$$

$\vartheta_\mu(x)$ is the gauge variable that takes on values in $[0,2\pi)$ and lives on the link (x,μ). The matrix Q in the fermion part of the action is identical to that in Eqs. (IV.1.4) and (IV.1.5). The Dirac indices α and β take on values 0 or 1 and there is only one "color."

The subgroup Z(200) was used to approximate U(1) in the pure gauge part of the action and the Metropolis algorithm was used to update the lattice. All our computation was done on a VAX 11/780 computer and took about 300 hours of CPU time. We worked on a 4×12 lattice at a coupling constant of β=2.5 and a hopping parameter value of κ=0.25. We chose to work at only one value of the coupling where the correlation length is of order unity due to limitations on computer time. The continuum limit of this model is not reached until the correlation length is very large [7]. Our only attempt has been to study the validity of various approximations in the pseudo fermion method in a region where it is essential to include the effects of dynamical fermions.

The mass gap of the model was calculated using the decay of the 2 point function:

$$<(\bar{\psi}\gamma_5\psi)_0 (\bar{\psi}\gamma_5\psi)_\tau> \quad . \tag{IV.2.10}$$

The operator $\bar{\psi}\gamma_5\psi$ has the correct quantum numbers to create, out of the vacuum, the meson whose mass we wish to measure. Such observables can be expressed in terms of the fermion propagator in the external gauge field:

$$<\bar{\psi}_l \psi_m>_\psi = \int [d\bar{\psi}][d\psi]\bar{\psi}_l \psi_m e^{\sum_{ij}\bar{\psi}_i Q_{ij}\psi_j} \qquad \text{(IV.2.11)}$$

$$= \frac{d}{dQ_{lm}} \int [d\bar{\psi}][d\psi] e^{\sum_{ij}\bar{\psi}_i Q_{ij}\psi_j}$$

$$= \frac{d}{dQ_{lm}}(\det Q) = Q^{-1}_{ml} \det Q \quad .$$

Similarly,

$$<\bar{\psi}_l \psi_m \bar{\psi}_n \psi_k>_\psi = (Q^{-1}_{ml} Q^{-1}_{kn} - Q^{-1}_{mn} Q^{-1}_{kl}) \det Q \quad , \qquad \text{(IV.2.12)}$$

and so the 2 point functions such as Eq. (IV.2.10) are obtained by running Gauss-Seidel inversions to find the various Q^{-1} elements. A typical result for this observable is shown in Fig. 4.1.

We first computed the mass gap in the quenched approximation where all closed fermion loops are neglected by setting the fermion determinant equal to a constant. Next, the "exact" calculation was done by including the effects of dynamical fermions via Gauss-Seidel as explained in Section 1. This method is exact to the extent that the Gauss-Seidel converges exponentially to the correct value of Q^{-1}. The results already tell us something important; the inclusion of dynamical fermions has a nontrivial effect which changes the mass by 25%. (Table 1).

We now turn to the pseudo fermion method. The pseudo fermion Monte Carlo was run only once for every gauge sweep through the lattice, i.e., the update of the entire gauge field configuration as in Eq. (IV.2.7). To see how well the determinant estimates from the pseudo Monte Carlo were converging, we

computed the determinant in the linear approximation, Eq. (IV.2.5) and compared it with the exact determinant (Eq. (IV.1.8)) evaluated at the same link in the same configuration. Some representative results are shown in in Fig. 4.2. We see that even after 100 pseudo sweeps the determinant estimate can still be off by ~50% from the correct value. We ascertained that our programs were correct by running for a very large number of pseudo Monte Carlo sweeps (up to 10,000) until the linearized result converged to the exact result for small hit size.

Table 1: Mass Gap of the Schwinger Model

Algorithm	Number of pseudo-sweeps	Mass in units of $\frac{1}{a}$
Quenched	-	0.708 ± 0.041
Exact (G.S.)	-	0.986 ± .052
Pseudo-Fermion	100	1.020 ± 0.029
	50	1.059 ± 0.023
	25	1.000 ± 0.027
	12	1.040 ± 0.026
	6	0.925 ± 0.035

The mass gap, determined from runs in which the dynamical fermions are included via the auxiliary, pseudo field, is also given in Table 1. To make the error introduced by the linearization of the determinant small (Eq. (IV.2.7), "systematic" error), we chose the Metropolis hit size, $\delta\vartheta$, to be small. $\delta\vartheta$ was taken to be $\pm.05\cdot 2\pi$. This is 4 times smaller than the optimal[2] value used for the

2. The "optimal" value for the hit size is defined to be that value which gives an acceptance of 50%, i.e., the link move is accepted 50% of the time.

exact calculation. Since the hit size (the step size of the random walk) is smaller, the Monte Carlo using the pseudo field, for a given number of gauge sweeps, explores configuration space more slowly than the exact Monte Carlo. The CPU time per sweep is, however, much lower. The determinant estimates only converge as the number of pseudo sweeps, N, goes to infinity, so we have done runs of various choices: N = 6,12,25,50 and 100. The results are given in Table 1 and are plotted in Fig. 4.3.

We find our results to be quite encouraging. Even though the error in the determinant (or more generally, in the action) due to the finite statistics is often large (Fig. 4.2), the validity of the approximation is finally to be judged by the error on the mass gap (or other observables). It seems that already at N =12, the mass has converged to the asymptotic value. The masses from N=12, 25, 50, 100 are essentially constant (within statistical errors). A linear extrapolation of these 4 values is shown in Fig. 4.3 and gives the value, for the mass gap from the pseudo fermion method, of:

$$m_{pseudo} = 1.034 \pm .037 \ ,$$

which is consistent with the value obtained from the exact algorithm:

$$m_{exact} = .986 \pm .052 \ .$$

When the dynamical fermions are "turned on", the exact calculation shows us that the mass gap shifts by approximately 25%. The pseudo fermion algorithm is able to reproduce this shift and the two methods agree to within 5%.

1. Why did it work so well?

It is actually quite surprising that as few as 12 to 25 sweeps in the pseudo field can give such (apparently) good results. As was mentioned in the last section, the determinants are not very well determined for N=12,25, with 50-100% errors being common. So, how can it work? The answer is that the gauge field is moving through configuration space slowly - the (true) determinant ratio at a particular link does not change very rapidly as the Monte Carlo runs. This means that the wildly fluctuating determinant estimates tend to average out over gauge sweeps, effectively giving a much more accurate estimate of the determinant.

One realizes that it is not the value of N that matters - rather, it is the ratio of speeds through configuration space of the two Monte Carlos that counts. The pseudo field, φ, must explore its configuration space much more rapidly than the gauge field does.

The argument so far has been intuitive. One can make it more precise by considering the Langevin formulation of the problem. The Langevin equations are a set of stochastic differential equations (i.e. containing noise terms) which are equivalent to the path integral. One can think of them in terms of a Monte Carlo simulation - in Appendix 3 it is shown that the Langevin method is just another method, similar to the Metropolis procedure, of satisfying detailed balance. For our purposes, the usefulness of this formalism lies in the fact that it is an easy way to write down equations which describe what the Monte Carlo is doing, and it also explicitly contains the ratio of speeds through configuration space. The Langevin equations for our problem are:

$$\dot{U} = -\frac{\delta S^{gauge}}{\delta U} - <\overline{\varphi}_i \varphi_j>(\frac{\delta Q}{\delta U})_{ij} + \eta_U \qquad \text{(IV.3.1)}$$

$$\dot{\varphi} = \frac{1}{\tau}\overline{\varphi}Q + \frac{1}{\tau^{1/2}}\eta_\varphi \qquad \text{(IV.3.2)}$$

$$\tau \ll 1 \quad .$$

The condition $\tau \ll 1$ insures that the φ field moves through configuration space much faster than the U field. η_U and η_φ are gaussian random noise terms, satisfying:

$$<\eta_U(t')\eta_U(t)> = <\eta_\varphi(t')\eta_\varphi(t)> = 2\delta(t'-t) \quad , \tag{IV.3.3}$$

which serves to normalize them. The "time", t, is a fictitious time and is the analog of the Monte Carlo sweep number.

Think of these equations as describing a Monte Carlo evolution of the fields. The gaussian random noise terms are the analogs of the "random hit", δU, of the Metropolis procedure. The other terms on the right hand sides are $-\frac{\delta S}{\delta U}$ and $-\frac{\delta S_{pseudo}}{\delta \varphi}$: they act to bias the evolution of the field toward field configurations with low action and are the analog of the acceptance, rejection part of the Metropolis procedure.

The term $<\bar{\varphi}_i \varphi_j>(\frac{\delta Q}{\delta U})_{ij}$ is the contribution of the fermion determinant. Define $S_{fermion}$ through,

$$\det Q = e^{tr \ln Q} = e^{S_{fermion}} \quad , \tag{IV.3.4}$$

and so,

$$\delta S_{fermion} = \delta tr \ln Q = tr Q^{-1} \delta Q = <\bar{\varphi}_i \varphi_j> \delta Q_{ij} \quad . \tag{IV.3.5}$$

The brackets in $<\bar{\varphi}_i \varphi_j>$ represent a "time" average over an interval much longer than the φ evolution time scale, τ; it is the analog of the average over N pseudo sweeps in the usual Monte Carlo approach. It is shown in Appendix 3 that τ corresponds to $\frac{(\delta U)^2}{N}$ of the Monte Carlo.

We now come to our claim: for τ sufficiently small, the brackets in $<\overline{\varphi}_i\varphi_j>$ can be completely removed from Eq. (IV.3.1), which is equivalent to setting $N=1$ in the Monte Carlo.

The intuitive justification for this is the following. τ small implies that, on the evolution time scale of the φ's, the U field is static. The factor $\frac{\delta Q}{\delta U}$ is thus, essentially, a constant. This implies that, in the time evolution of the U field for k steps, the contribution of this term is,

$$(\sum_{kterms} \overline{\varphi}_i \varphi_j)\,(\frac{\delta Q}{\delta U})_{ij} \quad , \tag{IV.3.6}$$

and so the effect of the term in each time step is,

$$\frac{1}{k}(\sum_{kterms} \overline{\varphi}_i \varphi_j)\,(\frac{\delta Q}{\delta U})_{ij} = <\overline{\varphi}_i \varphi_j>\,(\frac{\delta Q}{\delta U})_{ij} \quad . \tag{IV.3.7}$$

We see that, for the averaging over gauge configurations to work, the Langevin equations must be linear in the brackets. If the brackets are removed from a quadratic term, $<\overline{\varphi}\varphi><\overline{\varphi}\varphi>$, the averaging produces the incorrect result, $<\overline{\varphi}\varphi\overline{\varphi}\varphi>$. A proof that the brackets can be removed has been given for a simple model [8]. For our model, we give numerical evidence that this is the case.

The above argument suggests that the correct theory is obtained even in the extreme limit: $N=1$, $\delta\vartheta\to 0$. To check this, we ran Monte Carlo's for $N=1$, $\delta\vartheta=10,7,5,3$ (measured in units of $\frac{2\pi}{200}$). The results are plotted in Fig. 4.4 as a function of $\delta\vartheta$. It appears that the results are indeed converging to the correct value as $\delta\vartheta\to 0$. We take these data to be a strong indication, at least for the Schwinger model, that the intuitive arguments given in this section are correct. The determinant estimates coming from a pseudo Monte Carlo with $N=1$ are, of course, very bad; the only way for the correct results to be found is if the

averaging mechanism is in fact working.

For their simple model, Fucito and Marinari [8] prove that the error in solving the Langevin equations (with the brackets removed) for finite τ is proportional to τ. For the Monte Carlo this would mean that the error in the pseudo fermion method should be proportional to $\frac{(\delta U)^2}{N}$. This is not what one might naively expect - for instance, the determinant estimates converge as $\frac{1}{\sqrt{N}}$, so we might expect the overall error to be the same. From Fig. 4.5, where the data are plotted vs $\frac{1}{\sqrt{N}}$ and $(\delta\vartheta)^2$, it seems that the error, for the Schwinger model, is also proportional to $\frac{(\delta U)^2}{N}$. Again, it is the averaging mechanism which is responsible for the error to be of higher order than naively expected.

Let us now summarize what we have learned from the Schwinger model calculation. We have seen that, for surprisingly small choices of N, the averaging mechanism implies that good results can still be obtained. The exciting possibility exists that a Monte Carlo, including the effects of dynamical fermions, can run at a speed which is only a factor N slower than the pure gauge calculation (and not go as volume squared!). This is somewhat misleading, however. As the correlation length, ξ, grows, the "systematic" error associated with updating many gauge links using "old" Q^{-1} estimates becomes more severe. This implies that δU will have to be made small, implying that the calculation moves through configuration space more slowly. So, the actual speed of the method is somewhat slower (how much?) than just a factor N over pure gauge. Let us now move on to the interesting case of SU(3) in 4 dimensions.

2. SU(3) in 4 dimensions

We have done some preliminary studies on the application of the pseudo fermion method to the real theory: SU(3) in 4 dimensions. The same strategy is adopted as in the Schwinger model calculation - we wish to test the reliability of the method by comparing against an exact algorithm based on Gauss-Seidel inversion. Due to the extreme slowness (12 VAX minutes per sweep) of this exact algorithm, we are forced to do our comparisons on very small, 2^4 lattices. We emphasize that it is only the exact algorithm which is limited to such small lattices - the pseudo fermion algorithm could easily be run on larger (i.e. 6^4) lattices, on our VAX 11/780. We want to make the finite size effects on our observable be the same, however. For this reason, both algorithms have been run on 2^4 lattices.

The observable which is compared is the average value of the plaquette, or 1×1 Wilson loop. This is, admittedly, a very simple observable. It does show the effects of the dynamical fermions, however, and can easily be obtained with high accuracy. The average plaquette serves as our indicator that the dynamical fermions are changing the distribution of gauge field configurations.

The work of this section is also described in [9]. SU(3) with dynamical fermions is presently being studied by Hamber, Marinari, Parisi and Rebbi [10]. They work with Susskind fermions and go up to 8^4 lattices. They do not compare with an exact algorithm. However, they do note that their average plaquette has the correct qualitative behavior as the dynamical fermions are included.

In this work, we again use the Wilson formulation of fermions on the lattice. We are presently coding the Susskind case and plan to study these and compare with the Wilson results. More will later be said about the advantages and disadvantages of these two competing formalisms.

The full group of SU(3) was simulated in the Monte Carlo. Though discrete subgroups of SU(3) exist, the largest one has been found to be too coarse for reliable calculations [11]. The SU(3) links are updated using a Metropolis procedure involving SU(2) subgroups [12]. The pure gauge part of the action was taken to be the usual Wilson form (Eq. (I.1.5)),

$$S_{plaq} = 1 - \frac{1}{6} tr\left(U_\mu(n)\, U_\nu(n+\mu)\, U_\mu^{-1}(n+\nu)\, U_\nu^{-1}(n) + h.c. \right) , \qquad \text{(IV.4.1)}$$

with the U's in the fundamental representation of SU(3), and β is related to the coupling by $\beta = \frac{6}{g^2}$. Since we use the Wilson formulation of fermions, on each site there are 12 (complex) degrees of freedom (4(spin)×3(color)). The action for the pseudo fields is quadratic in the fields. Since there exists an efficient algorithm for generating a random variable, x, distributed according to the probability distribution $P(x) \sim e^{-ax^2}$ [13], we updated the pseudo fields via a "heat bath" procedure [14]. This involves choosing the transition probabilities to be,

$$P_{k \to k+1} = e^{-S_{k+1}} , \qquad \text{(IV.4.2)}$$

so that Eq. (I.4.4) is immediately satisfied. When this procedure is possible it is much more efficient than the Metropolis procedure since successive configurations are much less correlated. The coding of this algorithm turned out to be quite a task. For example, there are 48 different Dirac $((1\pm\gamma^\mu)(1\pm\gamma^\nu))$ matrices needed for the next-nearest neighbor action of the pseudo field: $\bar{\varphi} Q^\dagger Q \varphi$

In order to avoid needless floating point operations (i.e. multiplication of zeros) these 48, 4×4 matrices were "hand wired" into the code. We worked in a coupling regime of β= 5.5 to 5.9 . The correlation length in this regime is approximately 1, and current calculations are being done in this regime.

Fig. 4.6 shows the plaquette for β= 5.5 to 5.9 . Along with the pure gauge

values, we plot the results from runs in which the fermions were included via the pseudo fermion method. We included 2 flavors of fermion with κ=.125 (for both). The points of Fig. 4.6 were obtained with the Metropolis hit size, δU, equal to the optimal pure gauge value (a "large" δU), and with N=20 . Already, we see that the qualitative effect is the correct one. With dynamical quarks included we expect the behavior of Wilson loops to change from an area law decay to a perimeter law decay. Physically, this corresponds to the fact that as one pulls two quarks apart, stretching the flux tube between them, a quark-antiquark pair can pop out of the vacuum, forming two mesons. This screening effect translates into perimeter decay of Wilson loops. If area law is changing to perimeter law, we would expect the expectations of loops to rise, and this is what is seen in Fig. 4.6.

Next, let us compare the pseudo fermion technique with the exact algorithm. For reasons of limited CPU time, this was done only at one value of the coupling, β=5.7 . Fig. 4.7 shows the results of the exact algorithm and the pseudo fermion results for N =20,40,80. δU is still equal to the large, optimal value. We see that N=20, is definitely too low, and that N=40,80 is converging to the true result. This was with the large δU; if δU is smaller, then, by the arguments given in the previous section, N can also be taken smaller.

As in the Schwinger model case, we investigated the limit N small, $\delta U \to 0$. Consider first the run with N=10, δU= large, optimal value. For this low value of N, the determinant estimates fluctuate wildly. Since δU is large, the determinant acts as a large random noise term and disorders the field configurations. This is seen in the run: the plaquette falls from the pure gauge value of .59 to $\sim$.40 . As δU is taken towards 0, we expect the averaging mechanism to cause the field configurations to become ordered and, eventually, give the correct result for the plaquette. This is what is seen. Fig. 4.8 shows the results for

N=10, $\delta U \to 0$. We see that the plaquette values are converging towards the correct answer of ~ .66 . The averaging mechanism is at work here, as it was for the Schwinger model.

We find our results to be quite encouraging. The good results obtained in the Schwinger model calculation do not seem to be model dependent. Even for the real theory, SU(3) in 4 dimensions, our results indicate that with a very modest number of pseudo sweeps, N~50, and with the extreme linearized approximation of updating all gauge links from the same pseudo Monte Carlo (Eq. (IV.2.7)), the correct distribution of field configurations is obtained.

Lastly, we would like to take a look at the systematic error we are making by using "old" determinant estimates for the update of the gauge links. What one would like to know is: how much does the "old" determinant deviate from the actual determinant, at each link. To see this, we did the following. Picking some typical gauge configuration, we held it fixed and found, via Gauss-Seidel, the determinants at each link and printed them out. Then, with the same configuration, we went back to the beginning and again found the determinants. This time, however, we allowed the links to move exactly as they do in a typical update. Taking the ratios of these two lists of determinants gives us the ratio of the true determinant divided by the "old" determinant, as a function of the link number through the sweep. Some results are shown in Fig. 4.9. We see that, through most of the sweep, the ratios are very close to 1 - the "old" determinant is the same as the true determinant. Though many of the links have been moved by δU, this apparently averages out - the determinant is a smooth function in configuration space. Towards the end of the sweep some sizable differences are found. However, there does not seem to be a systematic drift to values >1 or <1 as one might suspect.

Before concluding this chapter, we would like to say a few words about the choice of fermion formalism. In the Susskind approach, the components of the Dirac spinor are spread out among various sites, so there are less degrees of freedom per unit volume than in the Wilson approach. The advantages of this approach are:

(1) For the same lattice size, this method will run ~ 8 times faster than the Wilson method (in 4 dimensions).

(2) There is a remnant of continuous chiral symmetry on the lattice for $m=0$, at any g [15].

The disadvantages are:

(1) The unit cell in this formalism is $2a$ on a side, so, at least in weak coupling one needs 2^4 times the lattice volume as in Wilson.

(2) This formalism has species doubling problems. In 4 dimensions, there are 4 flavors. In [10], the square root of the determinant is used to effectively give 2 flavors. It is not clear that this procedure is correct.

The Wilson method has no species doubling problems, however, chiral symmetry is explicitly broken for any finite g. These two fermion formulations need to be studied further. We are presently coding the Susskind case and plan to compare it against the Wilson method in the future.

Conclusions

In closing, we should mention the prospects for implementing the fermion methods described in this chapter on parallel computers, in particular, those of the Homogeneous Machine ilk. There is no problem with the implementation of pseudo fermions since the interaction is local. Though it did turn out to be a second nearest neighbor interaction, this is no worse than the case of a pure

gauge theory (links interact via plaquettes) and is not a serious problem for the Homogeneous Machine. For the stochastic algorithm or the related hopping expansion method [16], the basic quantities needed are Wilson loops of arbitrary shapes. Using the algorithm described in Appendix 1, these loops can be calculated efficiently, with little extra effort on the part of the programmer. We feel that either method can be effectively implemented on the Homogeneous Machine.

With the combination of improved fermion techniques running on new, powerful computers, Monte Carlo calculations including dynamical quark effects on realistic lattice sizes will perhaps be possible in the near future.

References

[1] J. Kuti, Phys. Rev. Lett. 49, 183 (1982).

[2] F. Fucito, E. Marinari, G. Parisi and C. Rebbi, Nucl. Phys. B180 [FS3], 369 (1981).

[3] S. Otto, M. Randeria, Caltech preprint, CALT-68-991, accepted for publication in Nuclear Physics B.

[4] D. J. Scalapino and R. L. Sugar, Phys. Rev. Lett. 46, 519 (1981).

[5] D. Weingarten and D. Petcher, Phys. Lett. 99B, 333 (1981).

[6] C. B. Lang, H. Nicolai, Nucl. Phys. B200[FS4], 135 (1982).

[7] O. Martin, S. Otto, Nucl. Phys. B203, 297 (1982).

[8] F. Fucito and E. Marinari, Nucl. Phys. B190 [FS3], 266 (1981).

[9] S. Otto, Caltech preprint in preparation.

[10] H. Hamber, E. Marinari, G. Parisi, C. Rebbi, Brookhaven preprint, BNL-32456.

[11] G. Bhanot, Phys. Lett. 108B, 337 (1982).

[12] M. Okawa, Phys. Rev. Lett. 49, 353 (1982).

[13] G. Dahlquist, A. Bjorck, *Numerical Methods*, (Prentice-Hall, 1974), p. 453.

[14] M. Creutz, Phys. Rev. D21, 2308 (1980).

[15] C. Rebbi, Invited Talk presented at 19th Orbis Scientiae Meeting, January 1982.

J. Kogut, Les Houches and St. Andrews lectures, 1982, Illinois preprint, ILL-(TH)-82-46.

[16] A. Hasenfratz, Z. Kunszt, P. Hasenfratz and C. B. Lang, Phys. Lett 110B, 289 (1982) and Phys. Lett. 117B, 81 (1982).

Figure Captions

[4.1]A representative two-point function.

[4.2]Representative plots of the linearized and exact ratios of determinants, as functions of the change in angle $\delta\vartheta$, measured on a randomly chosen link in four different configurations. The angle is measured in units of $\frac{2\pi}{200}$. The curve is the ratio obtained from the exact algorithm and the straight line is that obtained in the linear approximation from a pseudo Monte Carlo of $N=100$ sweeps with a Metropolis hit size of ± 10.

[4.3]The mass gap as a function of $\frac{1}{\sqrt{N}}$, where N is the number of pseudo-sweeps. The solid line is a linear extrapolation of the last four data points. Also shown are the results for the quenched and exact algorithm, plotted at $\frac{1}{\sqrt{N}} = 0$.

[4.4]The mass gap as a function of the Metropolis hit size, $\delta\vartheta$ (measured in units of $\frac{2\pi}{200}$),for a fixed number of pseudo-sweeps, N=1. The exact value is plotted at $\delta\vartheta$=0.

[4.5](a) The mass gap as a function of $\frac{1}{N}$. (b) The mass gap as a function of $(\delta\vartheta)^2$, for N=1. Again, the exact and quenched mass gap results are plotted at $\delta\vartheta$=0.

[4.6]The average plaquette for SU(3). The squares are the pure gauge results; the triangles are the results with the inclusion of dynamical quarks (2 flavors) via the pseudo fermion method, N=20.

[4.7]The average plaquette at β=5.7. Shown is the pure gauge result (square), the Gauss-Seidel result (cross), and the pseudo fermion results, plotted as a function of $\frac{1}{N}$.

[4.8]The average plaquette as a function of hit size. N is fixed at 10, and δU is taken towards 0. For convenience, the results are plotted versus the acceptance (fraction of Metropolis hits accepted). Small δU implies high acceptance, and as $\delta U \to 0$, the acceptance $\to 1$. The pure gauge and Gauss-Seidel results are plotted at acceptance equal to unity.

[4.9]The ratio of "new" to "old" determinants for typical configurations at β=5.7. The determinants themselves deviate from 1.0 by, typically, .06 .

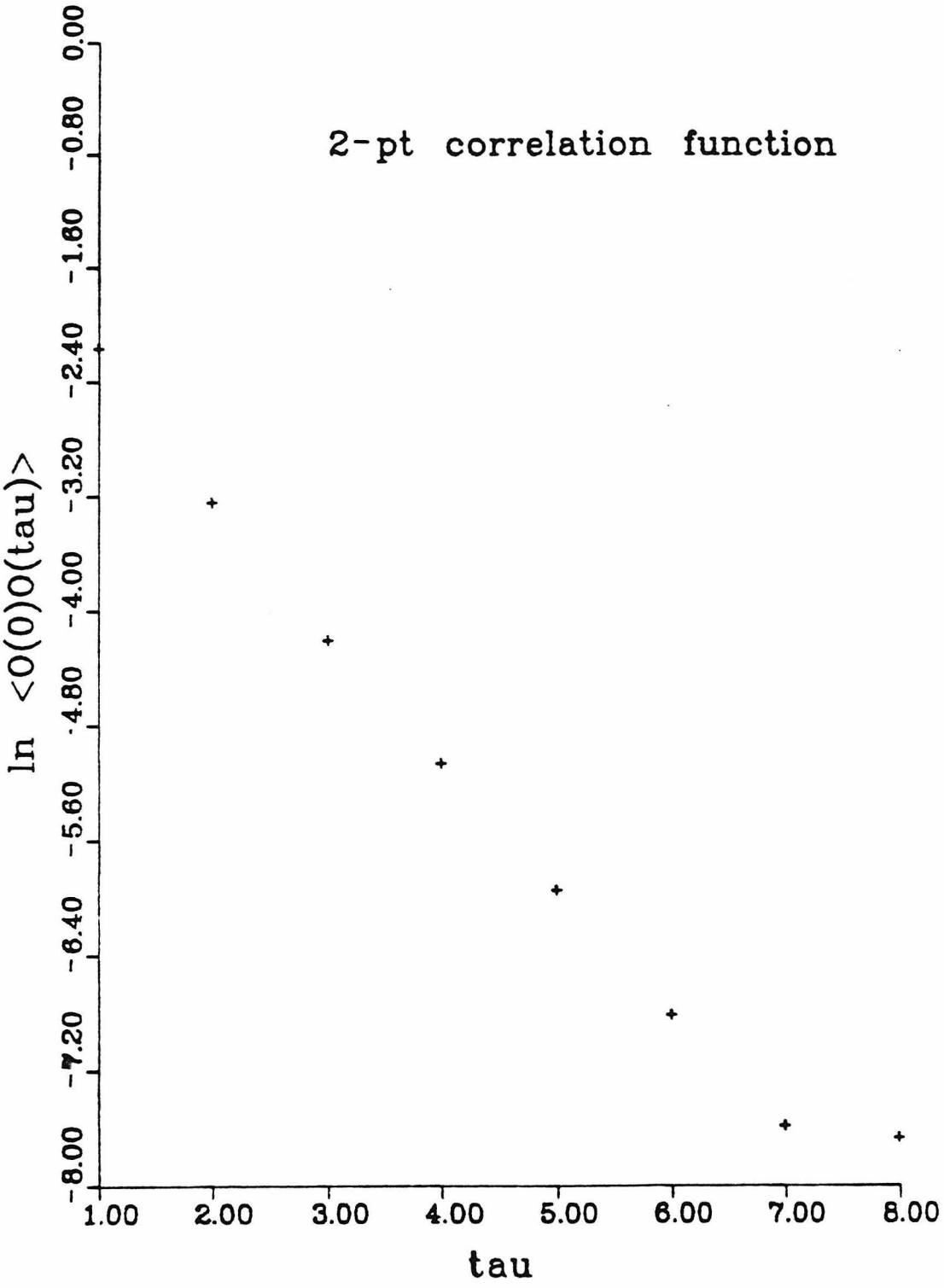

2-pt correlation function
ln <O(0)O(tau)>
0.00
-0.80
-1.60
-2.40
-3.20
-4.00
-4.80
-5.60
-6.40
-7.20
-8.00
1.00
2.00
3.00
4.00
5.00
6.00
7.00
8.00
tau

Fig. 4.1

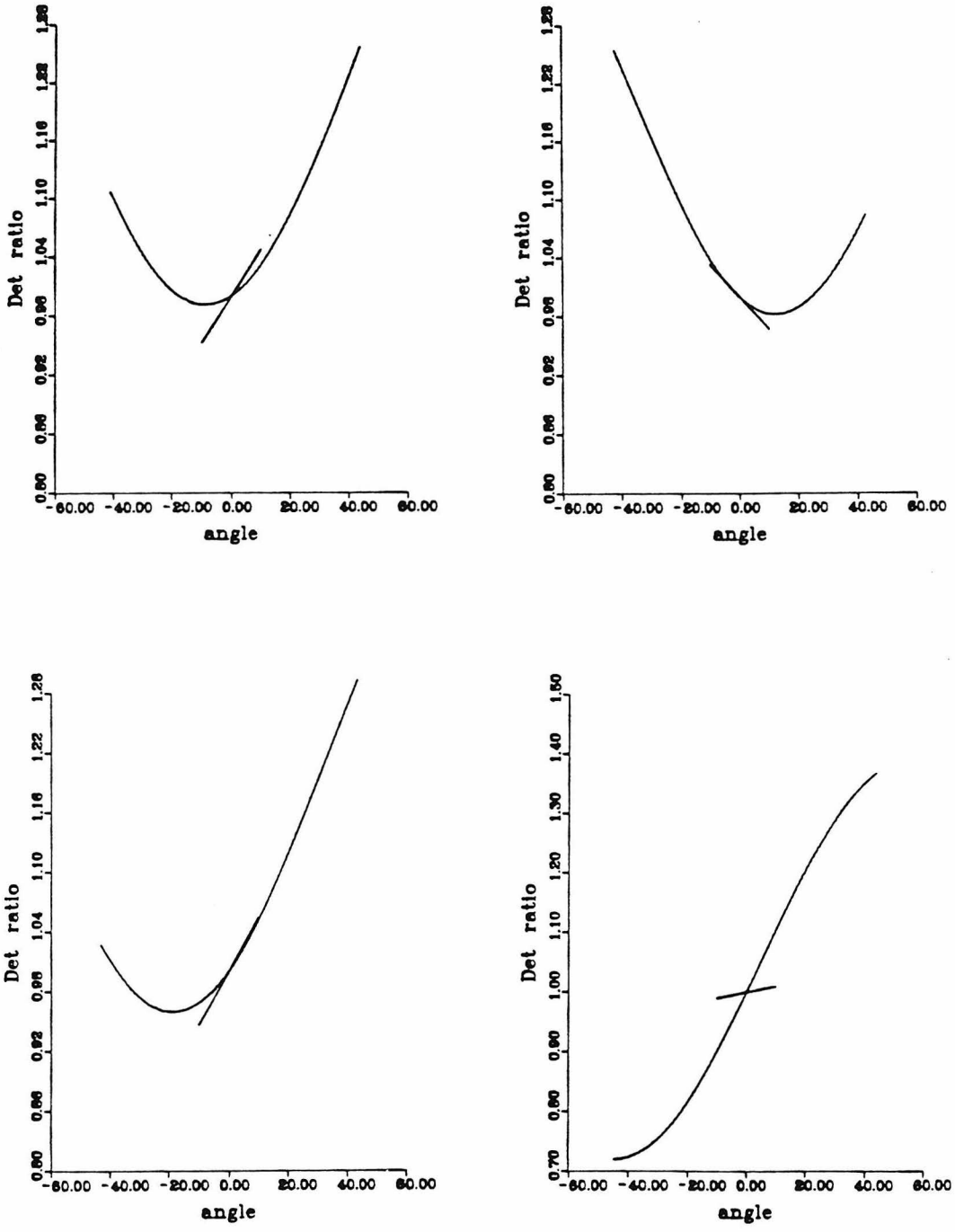

Det ratio
angle
-60.00 -40.00 -20.00 0.00 20.00 40.00 60.00
Det ratio
angle
-60.00 -40.00 -20.00 0.00 20.00 40.00 60.00
Det ratio
angle
-60.00 -40.00 -20.00 0.00 20.00 40.00 60.00
Det ratio
angle
-60.00 -40.00 -20.00 0.00 20.00 40.00 60.00

Fig. 4.2

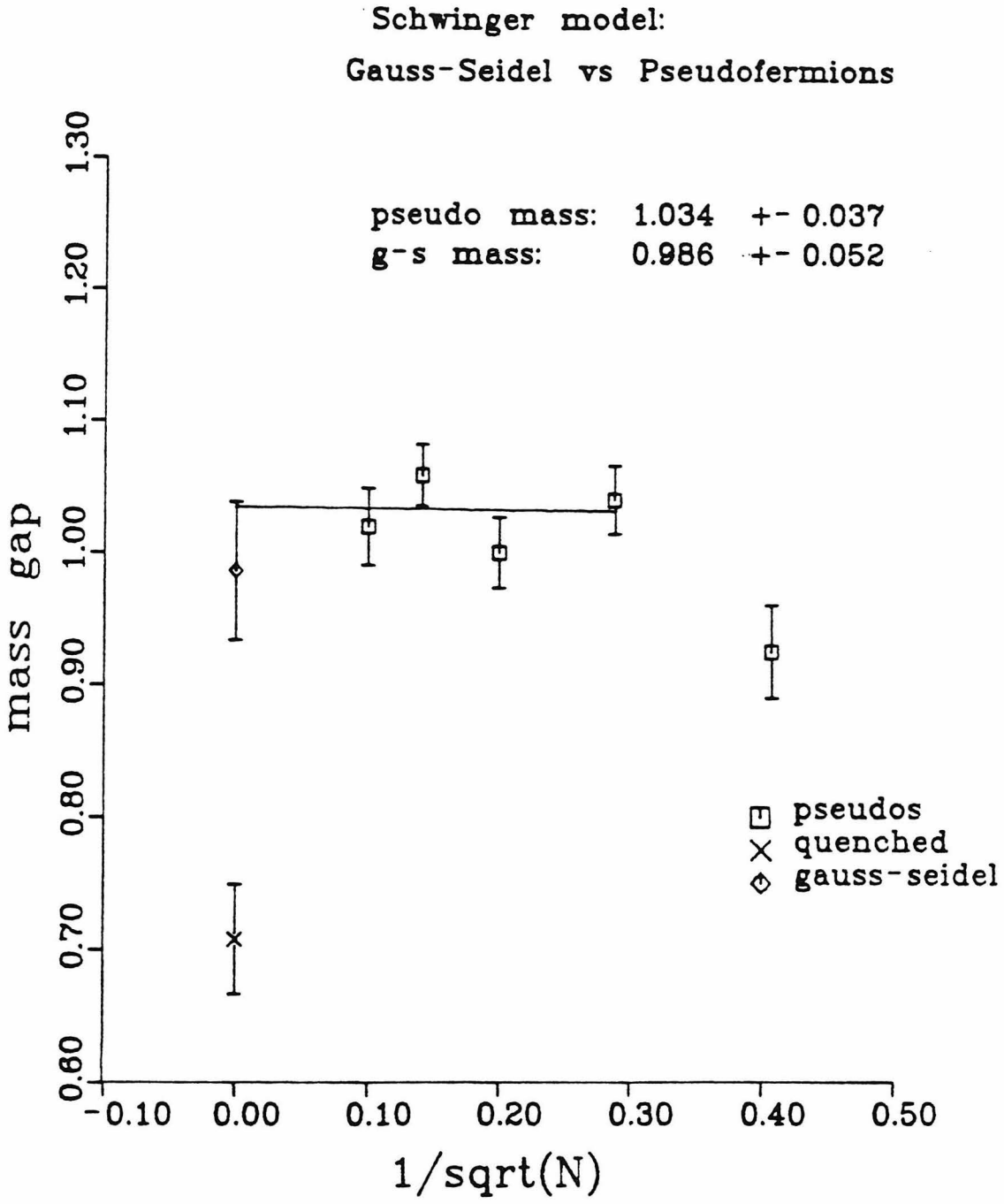

Schwinger model:
Gauss-Seidel vs Pseudofermions
pseudo mass: 1.034 +- 0.037
g-s mass: 0.986 +- 0.052
mass gap
1/sqrt(N)
pseudos
quenched
gauss-seidel
0.60
0.70
0.80
0.90
1.00
1.10
1.20
1.30
-0.10
0.00
0.10
0.20
0.30
0.40
0.50

Fig. 4.3

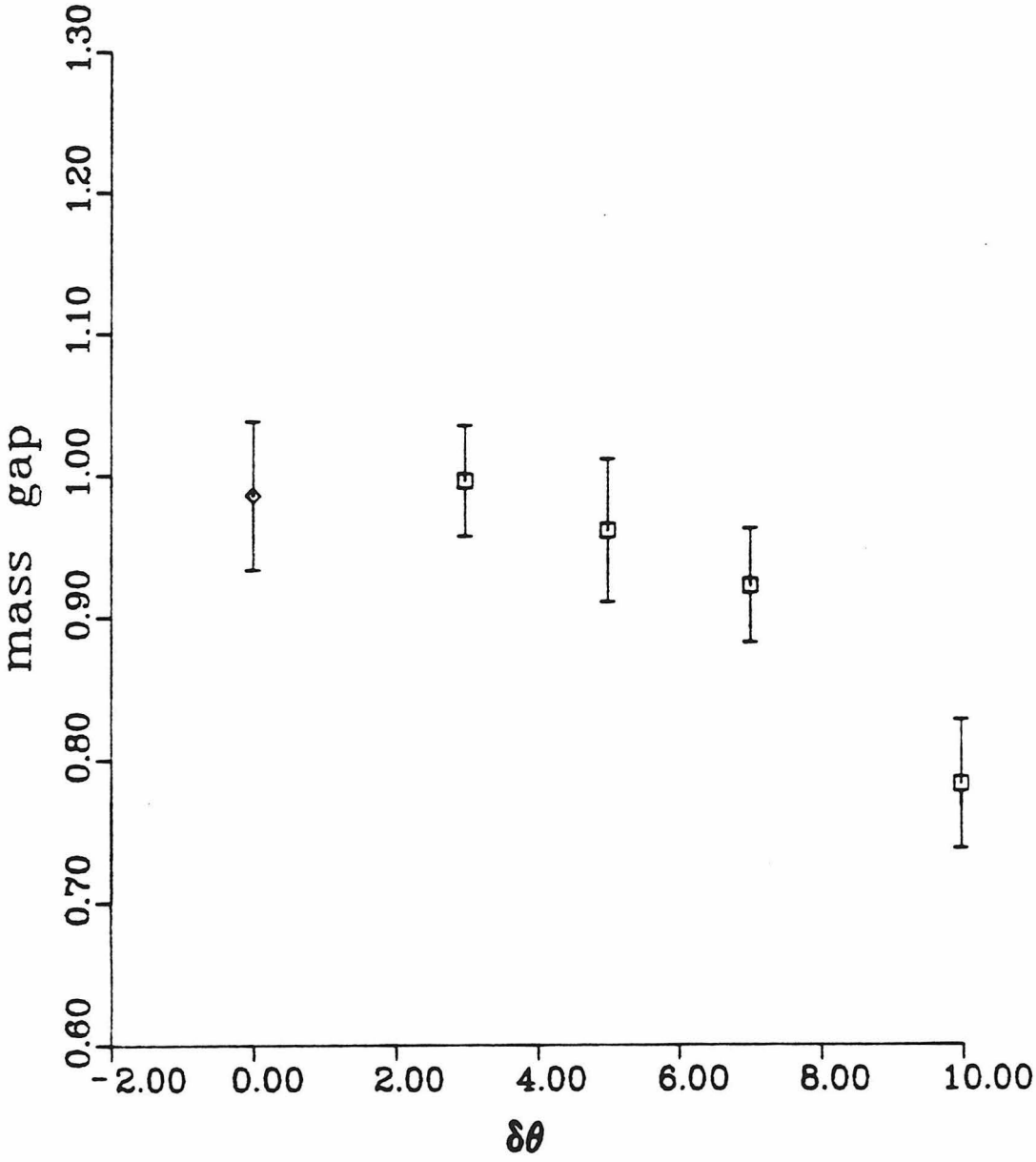

mass gap
0.60
0.70
0.80
0.90
1.00
1.10
1.20
1.30
-2.00
0.00
2.00
4.00
6.00
8.00
10.00
δθ

Fig. 4.4

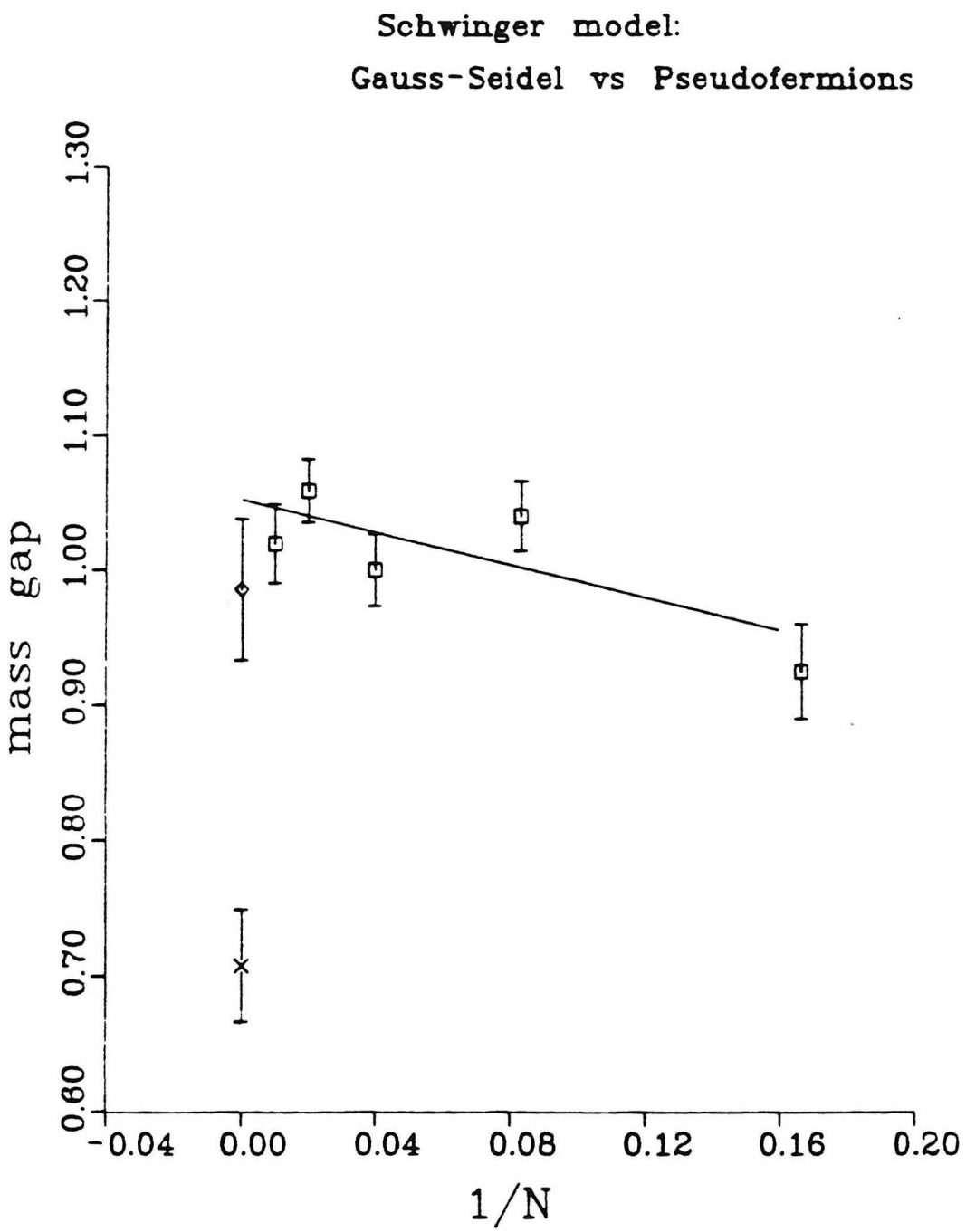

Schwinger model:
Gauss-Seidel vs Pseudofermions
mass gap
1.30
1.20
1.10
1.00
0.90
0.80
0.70
0.60
-0.04
0.00
0.04
0.08
0.12
0.16
0.20
1/N

Fig. 4.5a

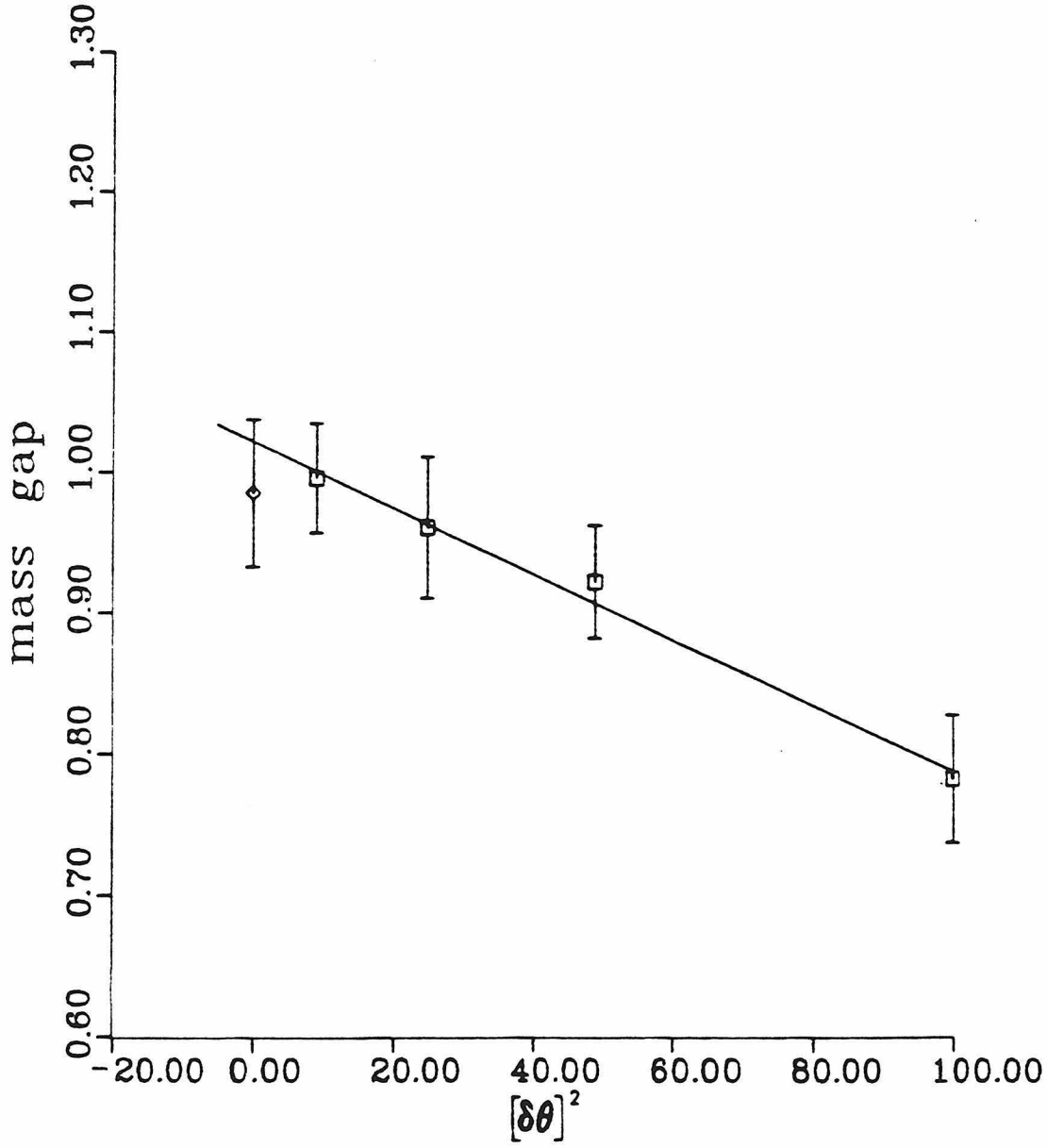

mass gap
0.60
0.70
0.80
0.90
1.00
1.10
1.20
1.30
-20.00
0.00
20.00
40.00
60.00
80.00
100.00
$[\delta\theta]^2$

Fig. 4.5b

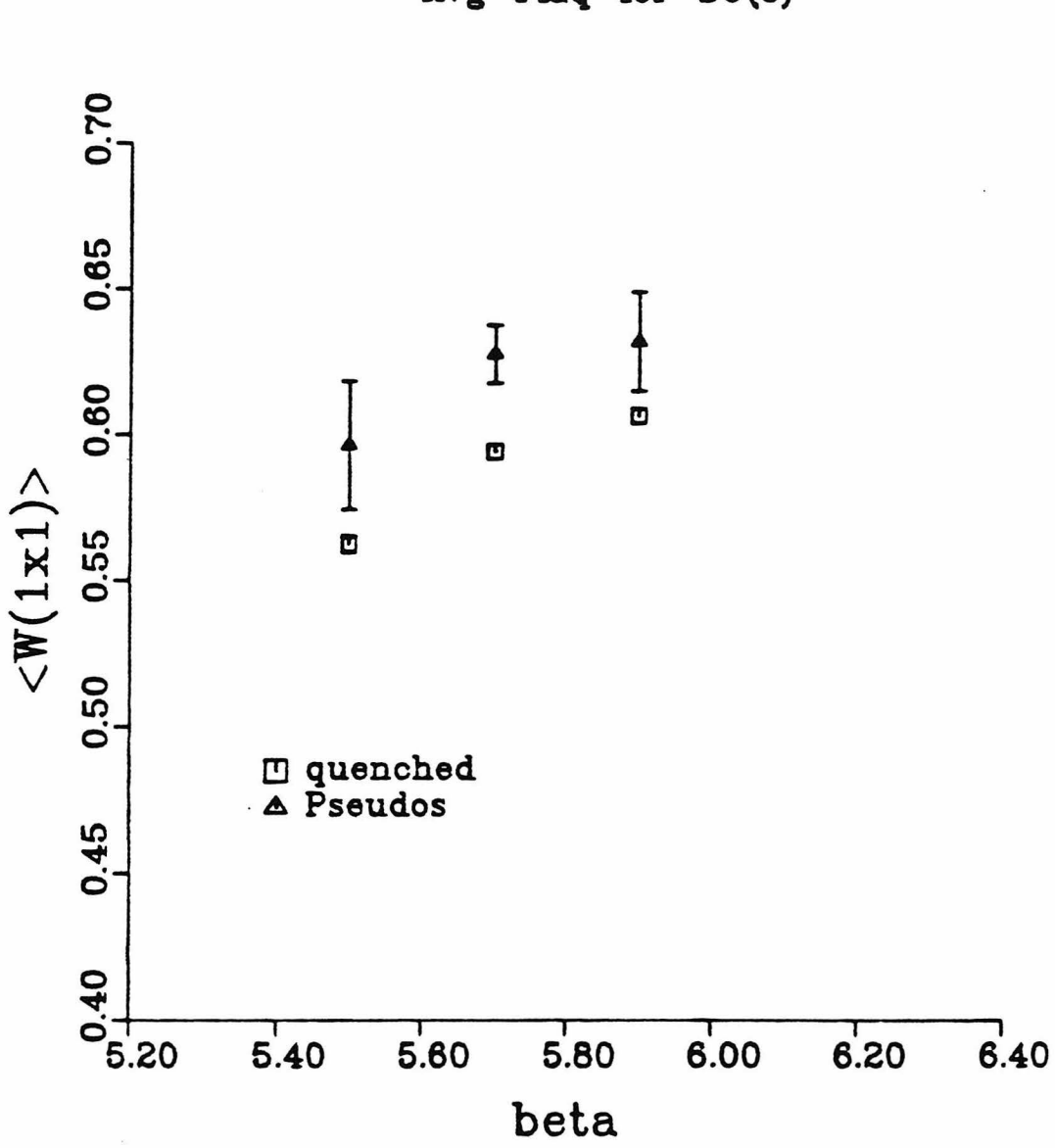

Avg Plaq for SU(3)
<W(1x1)>
0.40
0.45
0.50
0.55
0.60
0.65
0.70
5.20
5.40
5.60
5.80
6.00
6.20
6.40
beta
quenched
Pseudos

Fig. 4.6

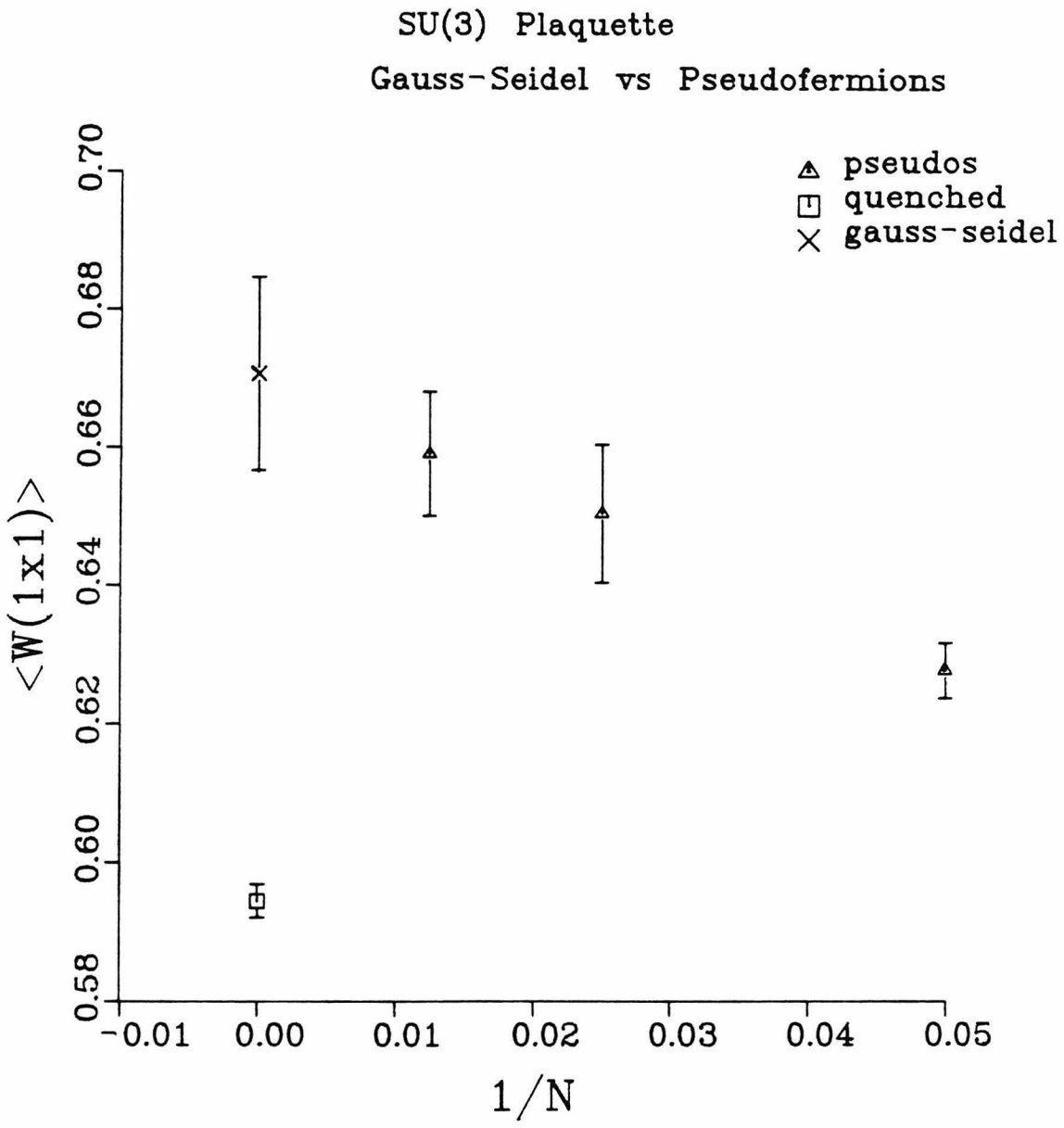

SU(3) Plaquette
Gauss-Seidel vs Pseudofermions
pseudos
quenched
gauss-seidel
<W(1x1)>
0.58
0.60
0.62
0.64
0.66
0.68
0.70
-0.01
0.00
0.01
0.02
0.03
0.04
0.05
1/N

Fig. 4.7

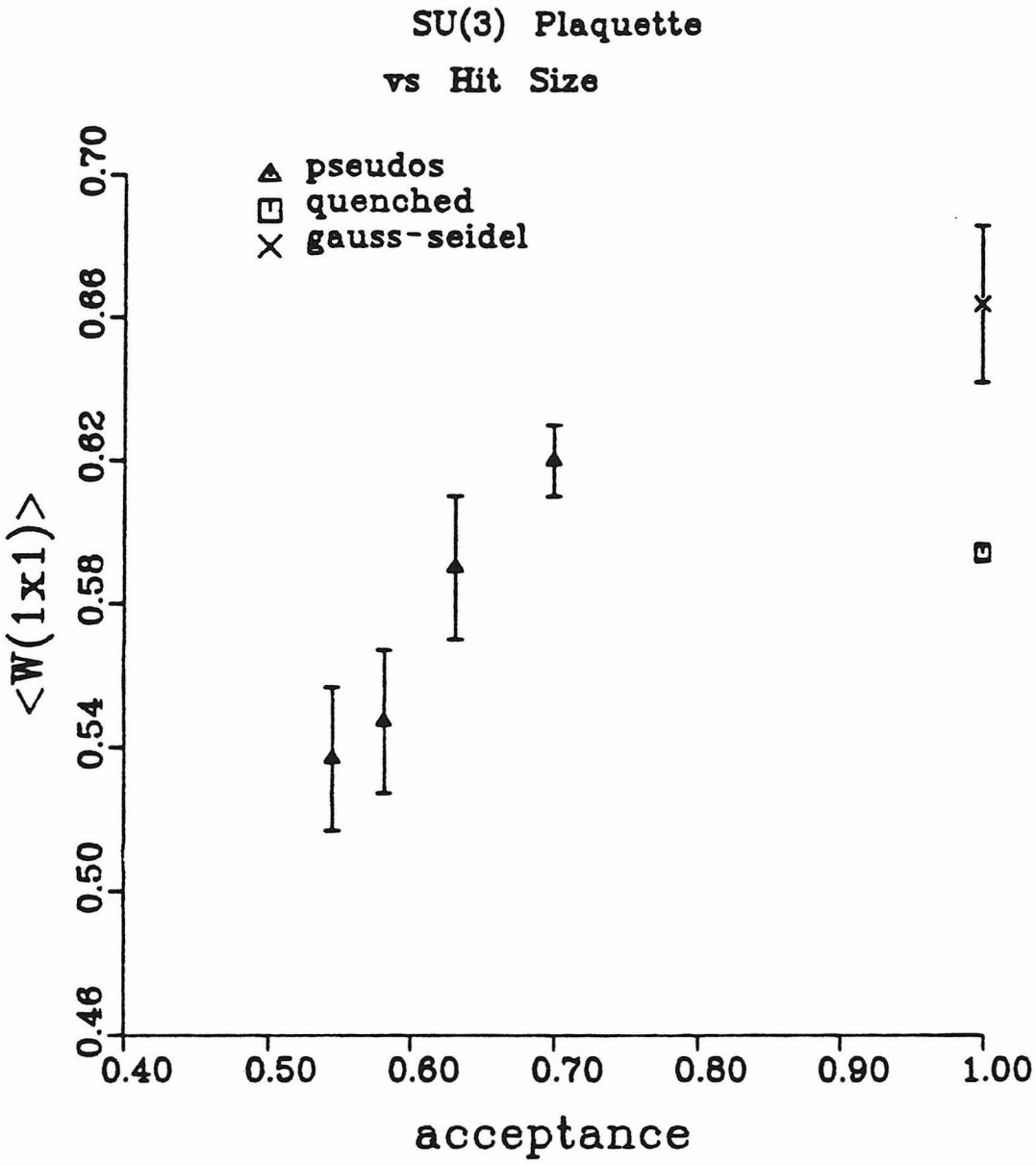
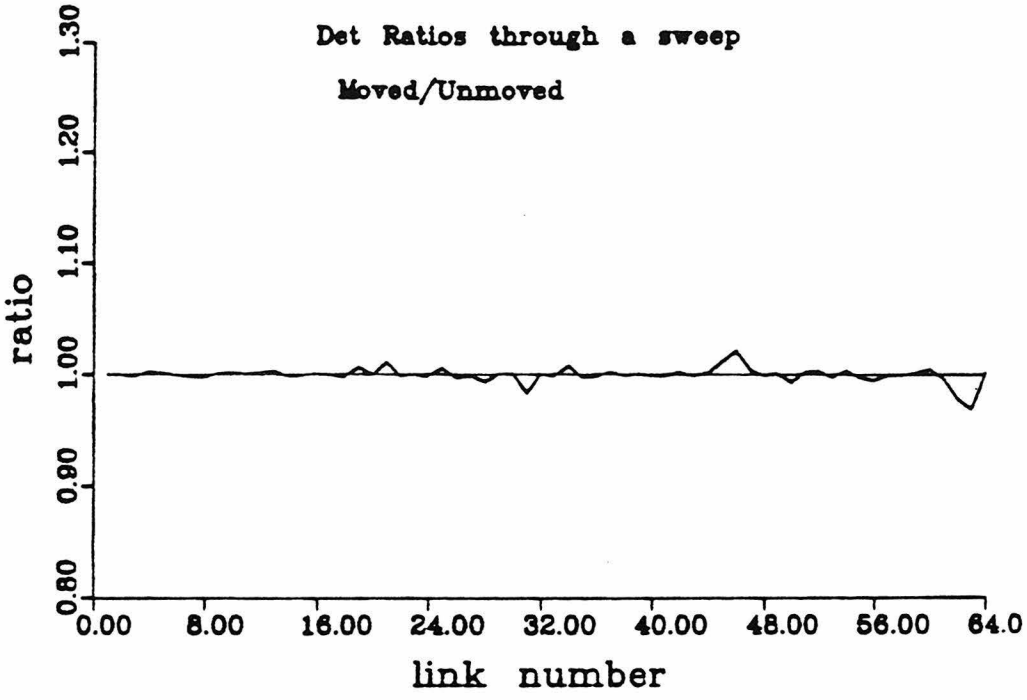

SU(3) Plaquette
vs Hit Size
pseudos
quenched
gauss-seidel
<W(1x1)>
0.48
0.50
0.54
0.58
0.62
0.66
0.70
0.40
0.50
0.60
0.70
0.80
0.90
1.00
acceptance

Fig. 4.8


Det Ratios through a sweep
Moved/Unmoved
ratio
1.30
1.20
1.10
1.00
0.90
0.80
0.00
8.00
16.00
24.00
32.00
40.00
48.00
56.00
64.0
link number

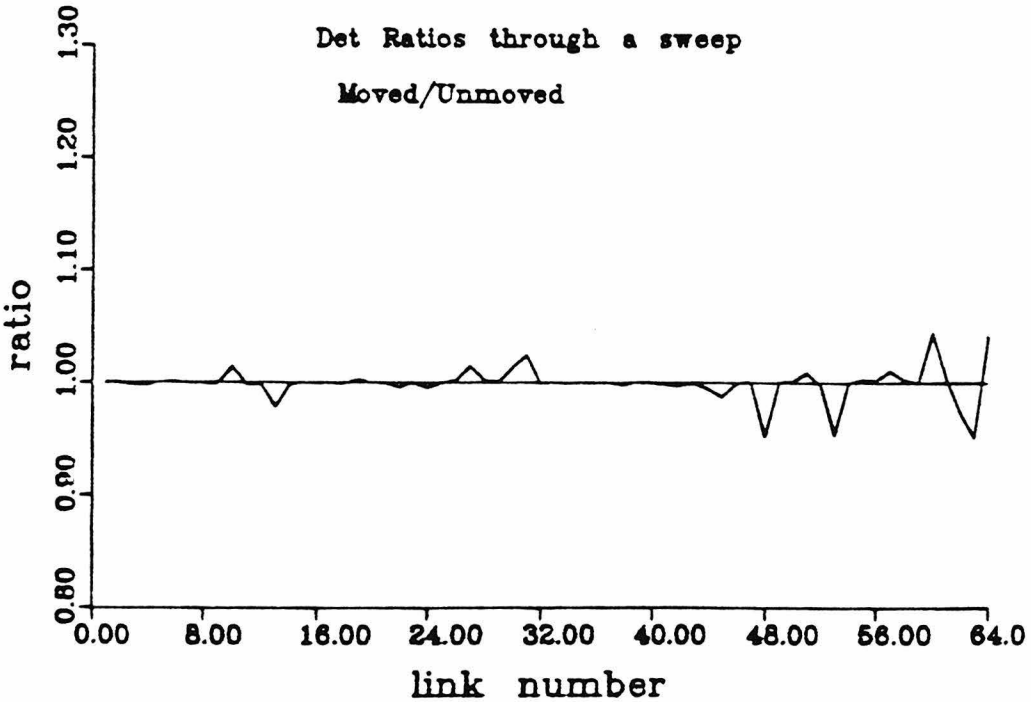

Det Ratios through a sweep
Moved/Unmoved
ratio
1.30
1.20
1.10
1.00
0.90
0.80
0.00
8.00
16.00
24.00
32.00
40.00
48.00
56.00
64.0
link number

Fig. 4.9

Appendix 1: Wilson Loops on the Homogeneous Machine

At first sight, the calculation of Wilson loops of arbitrary shape and orientation on the Homogeneous Machine seems very difficult. A loop such as that shown in Fig. 2.6(d) can intersect the subcells of several processors. Keeping track of all the necessary communications for increasingly complex loops (which we need for glueball calculations) is a difficult task for the programmer.

This turns out not to be the case, however, if one sets up the algorithm in the following way. Describe the shape of a loop by strings of integers, where each integer corresponds to a unit translation vector of the lattice: 1 corresponds to $+\hat{x}$, 2 corresponds to $+\hat{y}$,..., 5 corresponds to $-\hat{x}$, etc. For example, the loop of Fig. 2.6(d) is described by the string "253617". A routine is then constructed which takes the starting location of the loop and a string of integers (of arbitrary length) as input,and produces the matrix product along that path. This is done by moving to the site of the lattice where the loop starts and then reading the input string one integer at a time. When an integer is read, the matrix in the corresponding direction is fetched, multiplied into the current matrix product and, finally, the current site location is incremented in the same direction. The routine is then ready to read the next integer. It continues this way, literally "walking around" the path described by the input string.

The only modification necessary for this algorithm to work on the Homogeneous Machine is a simple test which, at each step of the "walk," tests to see if one is stepping out of the current subcell. If this is true, the processor sends its current matrix product to the processor in the direction of the step. The string does not need to be passed. All the processors are calculating the same shape loop located at the same point in each subcell (the calculation is, again, synchronized by the communications procedure), so all processors will be at the same step of the same path.

The above algorithm for Wilson loops was quite easy to implement and once this was done, the code for glueball masses, including an arbitrary number of operators, required very little additional work. One slight complication is the fact that a 2-point correlation, being a global observable, cannot be easily calculated within the Homogeneous Machine. What we do is the following. First, all the Wilson loops on the lattice are calculated in the nodes of the machine. The zero momentum operators are then found by adding the loops in the spatial directions - this is done by adding results and passing the results towards processor 0. The zero momentum operators, for each time slice, now reside in processor 0 and all that needs to be done is for the 2-point correlations between the various slices to be calculated. Instead of doing this in processor 0, we pass the numbers to the Intermediate Host (IH) and have it calculate the correlations. This frees the array to continue on to the next sweep. Since the amount of work that the IH has to do is small in comparison to that required to sweep through a subcell, the IH has no trouble keeping up with the array.

Appendix 2: Gauss-Seidel Inversion

Consider the equation,

$$M\boldsymbol{x} = \boldsymbol{b} \tag{A2.1}$$

M is a matrix, $\boldsymbol{x}$ and $\boldsymbol{b}$ are column vectors. We want to solve Eq. (A2.1) for $\boldsymbol{x}$. The Gauss-Seidel method consists of iteratively solving the ith equation for x_i . One starts by picking some vector, $\boldsymbol{x}^{(0)}$, as an initial guess. Then go to the first row and solve for x_1 , go the second row and solve for x_2 (using the new value for x_1), and so on, sweeping down the rows of M to get the first iterate, $\boldsymbol{x}^{(1)}$. This procedure is continued until $\boldsymbol{x}$ is no longer changing. We can be more precise.

Write M as,

$$M = D - L - U \ , \tag{A2.2}$$

where D is the diagonal part of M, and L and U are the lower and upper triangular parts of M. The iteration described above can be written as:

$$(D-L)\boldsymbol{x}^{(k+1)} = \boldsymbol{b} + U\boldsymbol{x}^{(k)} \tag{A2.3}$$

or,

$$\boldsymbol{x}^{(k+1)} = (D-L)^{-1}\boldsymbol{b} + (D-L)^{-1}U\boldsymbol{x}^{(k)} \tag{A2.4}$$

$$= (D-L)^{-1}\boldsymbol{b} + (D-L)^{-1}U[(D-L)^{-1}\boldsymbol{b} + (D-L)^{-1}U\boldsymbol{x}^{(k-1)}]$$

etc., leading to,

$$\boldsymbol{x} = \{(D-L)^{-1} + (D-L)^{-1}U(D-L)^{-1} + (D-L)^{-1}U(D-L)^{-1}U(D-L)^{-1} + \cdots\}\boldsymbol{b}$$

or,

$$\boldsymbol{x} = (D-L)^{-1}\{1 + U(D-L)^{-1} + U(D-L)^{-1}U(D-L)^{-1} + \cdots\}\boldsymbol{b} \ . \tag{A2.6}$$

To see when this method converges, expand b in terms of the eigenvectors of $U(D-L)^{-1}$:

$$b = \sum_i \alpha_i \varphi_i \quad , \tag{A2.7}$$

where,

$$U(D-L)^{-1}\varphi_i = \lambda_i \varphi_i \quad . \tag{A2.8}$$

Then Eq. (A2.6) implies,

$$x = (D-L)^{-1} \sum_i \alpha_i [\, 1 + \lambda_i + \lambda_i^2 + \cdots]\varphi_i \tag{A2.9}$$

$$= (D-L)^{-1} \sum_i \alpha_i \frac{1}{1-\lambda_i} \varphi_i \quad .$$

We see that the Gauss-Seidel method converges if all the λ_i satisfy

$$\lambda_i < 1 \quad . \tag{A2.10}$$

Eq. (A2.9) also shows that the convergence goes as λ^N, where λ is the largest of the λ_i, and N is the number of iterations.

Appendix 3: Langevin equations are equivalent to the Path Integral

We will give a physicist's "proof" (non-rigorous) that the Langevin equations are equivalent to the usual path integral. It is based on going through the operations required to solve the equations. We will show that these operations satisfy detailed balance and hence that the correct ensemble of field configurations is obtained.

The path integral is,

$$Z = \int [d\varphi] e^{-S[\varphi]} \ , \tag{A3.1}$$

and observables are found through,

$$<f[\varphi]> = \frac{1}{Z} \int [d\varphi] f[\varphi] e^{-S[\varphi]} \ . \tag{A3.2}$$

Suppose we solve the set of stochastic differential equations:

$$\dot{\varphi}_i = -\frac{\partial S}{\partial \varphi_i} + \eta_i(t) \tag{A3.3}$$

$$<\eta_i(t)\,\eta_j(t')> = 2\delta_{ij}\,\delta(t-t') \ . \tag{A3.4}$$

Then the claim is:

$$<f[\varphi]> = \lim_{T\to\infty} \frac{1}{T} \int_0^T dt\, f[\varphi(t)] \ . \tag{A3.5}$$

The index i labels spacetime location. η is a gaussian noise, normalized by Eq. (A3.4). The "time", t, is a fictitious time and is the analog of the sweep number in a Monte Carlo.

Consider the numerical solution of Eq. (A3.3),(A3.4). We discretize "time" into steps of size ε. The equation for the evolution of the $(k+1)$th field from the kth becomes:

$$\varphi_i^{(k+1)} - \varphi_i^{(k)} = -\varepsilon \frac{\partial S}{\partial \varphi_i}|_{\varphi^{(k)}} + \varepsilon \sqrt{\frac{2}{\varepsilon}} R \quad . \tag{A3.6}$$

R is a gaussian random number of width 1, that is, the probability distribution for R, $P(R)$, is,

$$P(R) = \frac{1}{\sqrt{2\pi}} e^{-\frac{R^2}{2}} \quad . \tag{A3.7}$$

This distribution gives $<R^2>=1$. The distribution of all possible choices of $\varphi_i^{(k+1)}$ is shown in Fig. A3.1.

The condition of detailed balance reads:

$$n_k \, P_{k \to k+1} = n_{k+1} P_{k+1 \to k} \quad , \tag{A3.8}$$

where n_k is the number of configurations in state k and $P_{k \to k+1}$ is the transition probability from state k to state $k+1$. We want:

$$\frac{n_{k+1}}{n_k} = \frac{e^{-S_{k+1}}}{e^{-S_k}} \quad . \tag{A3.9}$$

So we need,

$$\frac{P_{k \to k+1}}{P_{k+1 \to k}} = e^{-(S_{k+1} - S_k)} = e^{-(\varphi^{(k+1)} - \varphi^{(k)}) \frac{\delta S}{\delta \varphi}|_{\varphi^{(k)}}} \quad . \tag{A3.10}$$

Verify this using the picture of the distribution in Fig. A3.1:

$$P_{k \to k+1} = C \exp\{ -\frac{1}{2 \cdot 2\varepsilon} [\varphi^{(k+1)} - (\varphi^{(k)} - \varepsilon \frac{\delta S}{\delta \varphi}|_{\varphi^{(k)}})]^2 \} \quad , \tag{A3.11}$$

and,

$$P_{k+1 \to k} = C \exp\{ -\frac{1}{2 \cdot 2\varepsilon} [\varphi^{(k)} - (\varphi^{(k+1)} - \varepsilon \frac{\delta S}{\delta \varphi}|_{\varphi^{(k+1)}})]^2 \} \quad ,$$

with C a constant. This gives,

$$\frac{P_{k\to k+1}}{P_{k+1\to k}} = \exp\{-\frac{1}{2\varepsilon}[\varepsilon(\varphi^{(k+1)}-\varphi^{(k)})\frac{\delta S}{\delta\varphi}|_{\varphi(k)} - \varepsilon(\varphi^{(k)}-\varphi^{(k+1)})\frac{\delta S}{\delta\varphi}|_{\varphi(k+1)} + O(\varepsilon^2)]\}.$$

Now,

$$\frac{\delta S}{\delta\varphi}|_{\varphi(k+1)} = \frac{\delta S}{\delta\varphi}|_{\varphi(k)} + (\varphi^{(k+1)}-\varphi^{(k)})\frac{\delta^2 S}{\delta\varphi^2}|_{\varphi(k)} + \cdots \qquad \text{(A3.12)}$$

$$= \frac{\delta S}{\delta\varphi}|_{\varphi(k)} + O(\varepsilon^{1/2}) \ .$$

We have, finally,

$$\frac{P_{k\to k+1}}{P_{k+1\to k}} = \exp\{ -(\varphi^{(k+1)}-\varphi^{(k)})\frac{\delta S}{\delta\varphi}|_{\varphi(k)} + O(\varepsilon^{1/2}) \} \ , \qquad \text{(A3.13)}$$

and so detailed balance is satisfied as $\varepsilon\to 0$.

This proves that the ensemble obtained through the solution of the Langevin equations has its members distributed according to the Boltzmann factor, e^{-S}, and hence it is equivalent to the path integral.

Finally, we would like to show that τ in the equations,

$$\dot{U} = -\frac{\delta S^{gauge}}{\delta U} - <\overline{\varphi}_i\varphi_j>(\frac{\delta Q}{\delta U})_{ij} + \eta_U \qquad \text{(A3.14)}$$

$$\dot{\varphi} = \frac{1}{\tau}\overline{\varphi}Q + \frac{1}{\tau^{1/2}}\eta_\varphi \ , \qquad \text{(A3.15)}$$

$$<\eta_U(t')\eta_U(t)> = <\eta_\varphi(t')\eta_\varphi(t)> = 2\delta(t'-t) \ ,$$

corresponds to $\frac{(\delta U)^2}{N}$ in the Monte Carlo.

The distance that the φ field travels through configuration space, which is proportional to $\sqrt{N}$ in the Monte Carlo, goes as $\frac{1}{\tau^{1/2}}$ in the Langevin approach since this multiplies the width of the distribution in Fig. A3.1. Hence,

$$\tau \propto \frac{1}{N} \ . \tag{A3.16}$$

To see the dependence on δU, shift the time scale by τ. Let $t \rightarrow t' = \frac{t}{\tau}$. The equations become,

$$\dot{U} = -\tau \frac{\delta S^{gauge}}{\delta U} - \tau <\overline{\varphi}_i \varphi_j> (\frac{\delta Q}{\delta U})_{ij} + \tau^{1/2} \eta_U \tag{A3.17}$$

$$\dot{\varphi} = \overline{\varphi} Q + \eta_\varphi \ , \tag{A3.18}$$

$$<\eta_U(t')\eta_U(t)> = <\eta_\varphi(t')\eta_\varphi(t)> = 2\delta(t'-t) \ .$$

We see that the Metropolis hit size, δU, is proportional to $\tau^{1/2}$ since this is, again, the width of the distribution. Therefore,

$$\tau \propto (\delta U)^2 \ ,$$

and we have,

$$\tau \propto \frac{(\delta U)^2}{N} \ . \tag{A3.19}$$

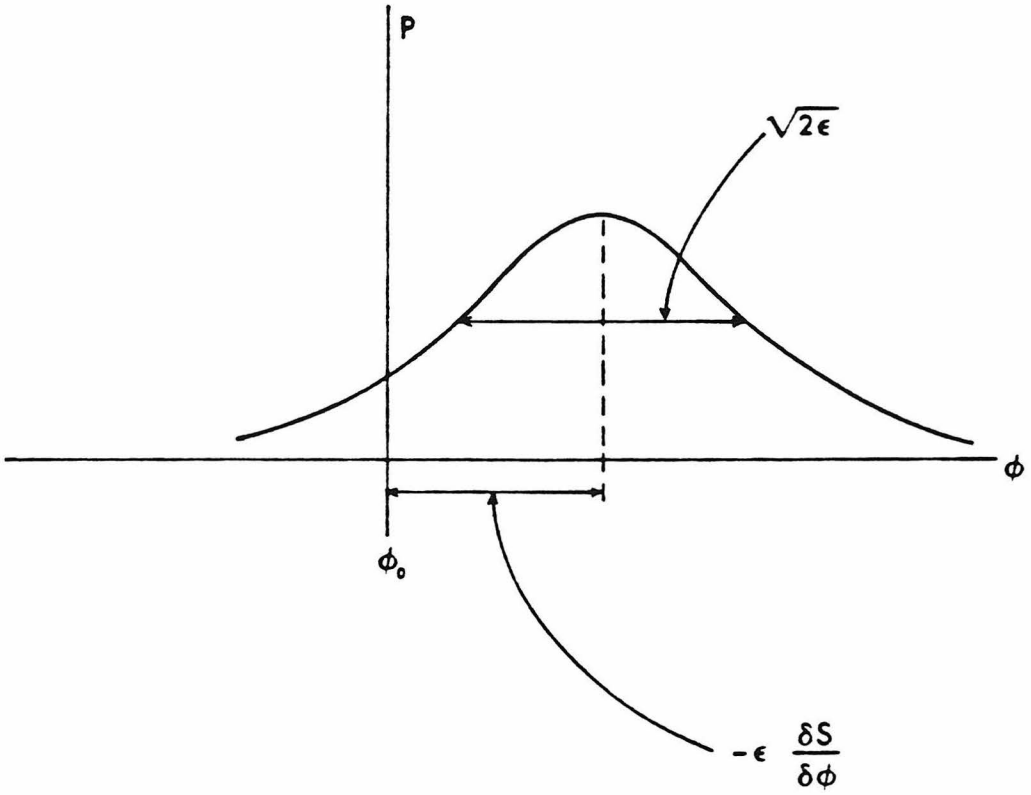


Fig. A.3.1