PHYSICAL REVIEW D VOLUME 40, NUMBER 6 15 SEPTEMBER 1989

Multigrid Monte Carlo method. Conceptual foundations

Jonathan Goodman*
Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, New York 10012

Alan D. Sokal†
Department of Physics, New York University, 4 Washington Place, New York, New York 10003
(Received 23 December 1988)

We present details of a stochastic generalization of the multigrid method, called multigrid Monte Carlo (MGMC), that reduces critical slowing down in Monte Carlo computations of lattice field theories. For Gaussian (free) fields, critical slowing down is completely eliminated. For a ϕ^4 model, numerical experiments show a factor of ≈ 10 reduction, over a standard heat-bath algorithm, in the CPU time needed to achieve a given accuracy. For the two-dimensional XY model, experiments show a factor of ≈ 10 reduction on the high-temperature side of criticality, growing to an unbounded reduction in the low-temperature regime. The algorithm is also applicable to nonlinear σ models, and to lattice gauge theories with or without bosonic matter fields.

I. INTRODUCTION

Monte Carlo computations of critical phenomena in statistical mechanics[1] and of the continuum limit in quantum field theory[2] have been greatly hampered by *critical slowing down*. In the traditional Monte Carlo algorithms, the autocorrelation time τ—that is, roughly speaking, the time needed to produce one "statistically independent" data point—grows rapidly as the critical point is approached, causing a corresponding rapid increase in the statistical error bars.[3] It is thus of considerable importance to devise new Monte Carlo algorithms having reduced critical slowing down.

In this paper we describe a new class[4] of Monte Carlo methods, called "multigrid Monte Carlo" (MGMC) methods, that should have shorter autocorrelation times in the critical region. For Gaussian (free) fields, a rigorous analysis (Secs. VII and VIII below) shows that critical slowing down is completely eliminated; the gain in efficiency over traditional algorithms thus grows without bound as the critical point is approached. For ϕ^4 fields, numerical experiments[4] show a gain in efficiency (measured in CPU units), over a single-site heat-bath algorithm, by a factor of ≈ 10. For the two-dimensional plane-rotator (XY) model, numerical experiments[5] show an efficiency gain of ≈ 10 on the high-temperature side of criticality (vortex regime), growing to an unbounded gain in the low-temperature (spin-wave) regime. (These numerical results are reviewed briefly in Sec. IX.) The MGMC idea applies to many other models, including nonlinear σ models and lattice gauge theories with or without bosonic matter fields. The performance of MGMC methods for these models is currently under investigation.[6,7]

The multigrid Monte Carlo method is a stochastic generalization of the multigrid (MG) method for solving finite-difference equations.[8] It is philosophically similar to, but technically quite different from, the block-spin renormalization group.[9] In the remainder of this introduction we review the causes of critical slowing down in the traditional Monte Carlo algorithms, and sketch the philosophy leading to MGMC methods. We also compare our work with other recent attempts at "collective-mode Monte Carlo"[10] methods, such as the Fourier-accelerated Langevin,[11] microcanonical, and hybrid[12] algorithms.

The critical slowing down of the traditional algorithms—such as the single-site Metropolis or heat-bath algorithms, or the unaccelerated Langevin method—arises fundamentally from the fact that their updates are *local*: in a single step of the algorithm, "information" is transmitted from a given site only to its nearest neighbors. Crudely one might guess that this "information" executes a random walk around the lattice. In order for the system to evolve to an "essentially new" configuration, the "information" has to travel a distance of order ξ, the (static) correlation length. One would guess, therefore, that $\tau \sim \xi^2$ near criticality. This guess is correct for the Gaussian model (free field);[13] in general, the correct statement is that $\tau \sim \xi^z$, where z is a dynamical critical exponent[14] that for most models of interest is close to 2 (Ref. 15). It follows that the computational work needed to get *one* "statistically independent" sample from a d-dimensional lattice of linear size L grows as $\sim L^d \xi^z \gtrsim \xi^{d+z}$ (Ref. 16). This is an enormous computational demand.

What is to be done? Our knowledge of the physics of critical slowing down tells us that the slow modes are the long-wavelength modes, if the updating is purely local. The natural solution is therefore to speed up those modes by *collective-mode* (nonlocal) updating.[10] It is necessary, then, to *identify physically* the appropriate collective modes, and to devise an *efficient computational algorithm* for speeding up those modes. These two goals are unfortunately in conflict; it is very difficult to devise collective-mode algorithms that are not so nonlocal that their computational cost outweighs the reduction in criti-

cal slowing down. Specific implementations of the collective-mode idea are thus highly model dependent. At least three such algorithms have been invented so far: Fourier acceleration,[11,12] the multigrid Monte Carlo[4-7] (MGMC) algorithm, and the Swendsen-Wang algorithm and its generalizations.[17-25] Fourier acceleration and the MGMC algorithm are very similar in spirit (though quite different technically). Their performance is thus probably qualitatively similar, in the sense that they probably work well for the same models and work badly for the same models. We discuss this in more detail, along with the advantages and disadvantages of each algorithm, in Sec. X.

The phenomenon of critical slowing down is not confined to Monte Carlo simulations: very similar difficulties were encountered long ago by numerical analysts concerned with the numerical solution of partial differential equations. An ingenious solution, now called the multigrid (MG) method, was proposed in 1964 by the Soviet numerical analyst Fedorenko:[26] the idea is to consider, in addition to the original ("fine-grid") problem, a sequence of auxiliary "coarse-grid" problems that approximate the behavior of the original problem for excitations at successively longer length scales (a sort of "coarse-graining" procedure). The local updates of the traditional algorithms are then supplemented by coarse-grid updates. To a present-day physicist, this philosophy is remarkably reminiscent of the renormalization group—so it is all the more remarkable that it was invented two years before the work of Kadanoff[27] and seven years before the work of Wilson.[28] After a decade of dormancy, multigrid was revived in the mid 1970s,[29] and was shown to be an extremely efficient computational method. In recent years, multigrid methods have become an active area of research in numerical analysis, and have been applied to a wide variety of problems in *classical* physics.[30-37] In this paper we wish to show how a stochastic generalization of the multigrid method can be applied to problems in *statistical* physics, and hence also Euclidean *quantum* physics.

This generalization is based on the strong analogy between solving lattice systems of equations (such as the discrete Laplace equation) and making Monte Carlo simulations of lattice random fields. Indeed, given a Hamiltonian $H(\phi)$, the deterministic problem is that of *minimizing* $H(\phi)$, while the stochastic problem is that of *generating random samples* from the Boltzmann-Gibbs probability distribution $e^{-\beta H(\phi)}$. The statistical-mechanical problem reduces to the deterministic one in the zero-temperature limit $\beta \to +\infty$.

Many (but not all) of the deterministic iterative algorithms for minimizing $H(\phi)$ can be generalized to stochastic iterative algorithms—that is, dynamic Monte Carlo methods—for generating random samples from $e^{-\beta H(\phi)}$. For example, the Gauss-Seidel algorithm for minimizing H and the heat-bath algorithm for random sampling from $e^{-\beta H}$ are very closely related. Both begin with a trial configuration $\phi^{(0)}$ and update this configuration by sweeping successively through the lattice, working on one site x at a time. The Gauss-Seidel algorithm updates ϕ_x so as to *minimize* the Hamiltonian $H(\phi) = H(\phi_x, \{\phi_y\}_{y \neq x})$ when all the other fields $\{\phi_y\}_{y \neq x}$ are held fixed at their current values. The heat-bath method gives ϕ_x a new *random* value (independent of the old value) chosen from the probability distribution $\exp[-H(\phi_x, \{\phi_y\}_{y \neq x})]$, with all the fields $\{\phi_y\}_{y \neq x}$ again held fixed. As $\beta \to +\infty$ the heat-bath algorithm approaches the Gauss-Seidel algorithm.

In this paper we show that the deterministic multigrid method has likewise a stochastic generalization, which we call the multigrid Monte Carlo (MGMC) method. We explain this generalization in detail in Sec. VII, but let us give the basic idea now. The deterministic multigrid algorithm for minimizing H [see (3.1)] involves Gauss-Seidel sweeps on a sequence of auxiliary coarse-grid problems H_l (l = grid index), together with operations for passing from grid l to grid $l-1$ and vice versa. The multigrid Monte Carlo algorithm for generating random samples from $e^{-\beta H}$ [see (7.11)] is *identical* in structure: the only difference is that the Gauss-Seidel sweeps for minimizing H_l are replaced by heat-bath sweeps for generating random samples from $e^{-\beta H_l}$. It is far from obvious that this algorithm is *correct*—i.e., that its equilibrium distribution is the desired distribution $e^{-\beta H}$—but it is true, as we show in Sec. VII. The proof is based on the idea of *partial resampling*: this is a generalization of heat-bath updating in which we focus on a set of field variables rather than only one, and the new values need not be independent of the old values. In the MGMC method the resampling is applied to the entire coarse-grid field.

In this paper we describe versions of the MGMC algorithm for free fields, $P(\phi)$ models, nonlinear σ-models, and lattice gauge theories with or without bosonic matter fields. (We do not yet know how to include dynamical fermions.)

In the free-field (Gaussian) case, the analogy between minimizing H and generating random samples from $e^{-\beta H}$ can be made precise by using the formalism of second quantization. In fact, we show that the autocorrelation time of any stochastic linear iteration (e.g., heat bath or MGMC) is *equal* to the relaxation time of the corresponding deterministic linear iteration (Gauss-Seidel or MG). In particular, we can analyze exactly the Gaussian MGMC algorithm—proving rigorously the absence of critical slowing down—as well as the Gaussian successive over-relaxation (SOR) Monte Carlo algorithm of Adler[38] and Whitmer.[39]

Let us emphasize that although the multigrid method and the block-spin renormalization group (RG) are based on very similar *philosophies*—dealing with a single length scale at a time—they are in fact *very different*. In particular, the conditional coarse-grid Hamiltonian employed in the MGMC method is *not* the same as the renormalized Hamiltonian given by a block-spin RG transformation. The RG transformation computes the *marginal*, not the conditional, distribution of the block means—that is, it *integrates* over the complementary degrees of freedom, while the MGMC method *fixes* these degrees of freedom at their current (random) values. Our conditional Hamiltonian is given by an explicit finite expression,

while the marginal (RG) Hamiltonian cannot be computed in closed form. The failure to appreciate these distinctions has led to much confusion in the literature; we return to this point in Sec. X.

Let us now outline the contents of this paper.

Section II is a pedagogical introduction to multigrid methods in the simplest case, namely the solution of deterministic linear systems of equations. We first review briefly one traditional iterative method for solving linear systems (the Jacobi method) and explain the physical origin of critical slowing down. Further information on the traditional iterative methods (Jacobi, Gauss-Seidel, and SOR) can be found in the excellent books of Varga,[40] Schwarz, Rutishauer, and Stiefel,[41] Wachspress,[42] Young and Gregory,[43] and Young,[44] among others. Next we define and explain the multigrid algorithm and discuss some of its ingredients. Of course, we can only scratch the surface of what is now an extremely active area of numerical-analysis research, and we strongly suggest that the reader consult other references to obtain a broader (and perhaps more balanced) perspective. An excellent introduction to multigrid methods is given by Briggs;[45] we also recommend the articles of Brandt[46] and Chaps. 2–4 of the book of Hackbusch.[30] More advanced or specialized topics can be found in various conference proceedings,[31–37] as well as in the remainder of the book of Hackbusch.[30] The reader who is already familiar with multigrid methods can skim lightly over Sec. II.

In Secs. III–V we extend the deterministic multigrid method to nonlinear systems of equations, as would arise from minimizing a nonquadratic Hamiltonian. We treat $P(\phi)$ models (Sec. III), nonlinear σ models (Sec. IV), and lattice gauge theories (Sec. V). Our main goal here is to present the *interpolation operators* for each type of model: these operators play an identical role in the deterministic (MG) and stochastic (MGMC) algorithms, and it is conceptually clearer to introduce them first in the deterministic case, without the additional complications of stochasticity. The reader who is anxious to get to the Monte Carlo case as quickly as possible is advised, on a first reading, to read Secs. II and III and then skip directly to Sec. VII.

In Sec. VI we review the basic principles of dynamic Monte Carlo methods, and define the quantities which measure the critical slowing-down. In Sec. VII we discuss the relation between deterministic algorithms for minimizing a Hamiltonian and stochastic algorithms for generating random samples from the Boltzmann-Gibbs distribution; we explain the concept of partial resampling; and we introduce the multigrid Monte Carlo (MGMC) algorithm and prove its correctness. In Sec. VIII we consider an important class of dynamic Monte Carlo methods—the stochastic linear iterations for Gaussian models—and we show how the behavior of the stochastic algorithm is completely determined by the behavior of the corresponding deterministic algorithm for solving linear equations. This proves the absence of critical slowing down for the Gaussian MGMC algorithm.

In Sec. IX we review the results of some numerical experiments[4] with the MGMC method on a two-dimensional ϕ^4 model, and give preliminary results of experiments in progress[5,6] on the two-dimensional XY and O(4) models. We also give heuristic arguments that the MGMC method should completely eliminate critical slowing down (except for a possible logarithm) in all asymptotically free theories with a continuous symmetry group. In Sec. X we compare the MGMC method to other proposals for "collective-mode Monte Carlo" methods, summarize our findings, and discuss prospects for the future.

In the Appendix we give the details of our heat-bath algorithm for ϕ^4 models.

The logical organization of this paper is summarized in Table I.

Subsequent papers in this series will apply the MGMC method to the two-dimensional plane-rotator (XY) model,[5] the two-dimensional O(4)-symmetric nonlinear σ model,[6] and the U(1) pure lattice gauge theory,[7] among others, and present rigorous convergence proofs for linear MG and the Gaussian MGMC algorithm.[47]

II. DETERMINISTIC MULTIGRID METHODS: LINEAR SYSTEMS

In this section we give a pedagogical introduction to multigrid methods in the simplest case, namely the solution of deterministic linear systems of equations. The reader who is already familiar with multigrid methods can skim lightly over this section. The reader who is anxious to get to the stochastic (Monte Carlo) case as quickly as possible is advised to read this section and the next one, and then skip directly to Sec. VII.

A. Traditional iterative methods and critical slowing down

Consider, for purposes of exposition, the lattice Poisson equation $-\Delta\phi = f$ in a region $\Omega \subset \mathbf{Z}^d$ with zero Dirichlet data. Thus, the equation is

$$(-\Delta\phi)_x \equiv 2d\phi_x - \sum_{x':|x-x'|=1} \phi_{x'} = f_x \tag{2.1}$$

for $x \in \Omega$, with $\phi_x \equiv 0$ for $x \notin \Omega$. Our goal is to devise a rapidly convergent iterative method for solving numerically the linear system (2.1) (Ref. 48).

This is a special case of the more general problem of solving a linear system

TABLE I. Logical structure of this paper.

	Free field	$P(\phi)$ model	σ model	Gauge theory
Deterministic (MG)	Sec. II	Sec. III	Sec. IV	Sec. V
Stochastic (MGMC)	Secs. VII, VIII	Secs. VII, IX A	Secs. VII, IX B, IX C	Secs VIII, IX C

$$A\phi = f \ , \tag{2.2}$$

where A is a given nonsingular matrix and f is an arbitrary vector. In order to solve (2.2) we shall consider first-order stationary linear iterations of the general form

$$\phi^{(n+1)} = M\phi^{(n)} + Nf \ , \tag{2.3}$$

where $\phi^{(0)}$ is an arbitrary initial guess for the solution. Obviously, we must demand at the very least that the true solution $\phi \equiv A^{-1}f$ be a fixed point of (2.3); imposing this condition for all f, we conclude that

$$N = (I - M)A^{-1} \ . \tag{2.4}$$

The iteration (2.3) is thus completely specified by its *iteration matrix* M. Moreover, (2.2)–(2.4) imply that the error $e^{(n)} \equiv \phi^{(n)} - \phi$ satisfies

$$e^{(n+1)} = Me^{(n)} \tag{2.5}$$

and hence

$$e^{(n)} = M^n e^{(0)} \ . \tag{2.6}$$

That is, the iteration matrix is the amplification matrix for the error. It follows easily that the iteration (2.3) is convergent for all initial vectors $\phi^{(0)}$ if and only if the spectral radius $\rho(M) \equiv \lim_{n\to\infty} \|M^n\|^{1/n}$ is < 1; and in this case the convergence is exponential with asymptotic rate at least $\rho(M)$, i.e.,

$$\|\phi^{(n)} - \phi\| \le Kn^p \rho(M)^n \tag{2.7}$$

for some $K, p < \infty$ (K depends on $\phi^{(0)}$).[49]

Now let us return to the specific system (2.1). One simple iterative algorithm arises by solving (2.1) repeatedly for ϕ_x:

$$\phi_x^{(n+1)} = \frac{1}{2d}\left[\sum_{x':|x-x'|=1} \phi_{x'}^{(n)} + f_x\right] . \tag{2.8}$$

(2.8) is called the *Jacobi iteration*. It is convenient to consider also a slight generalization of (2.8): let $0 < \omega \le 1$, and define

$$\phi_x^{(n+1)} = (1-\omega)\phi_x^{(n)} + \frac{\omega}{2d}\left[\sum_{x':|x-x'|=1} \phi_{x'}^{(n)} + f_x\right] . \tag{2.9}$$

(2.9) is called the *damped Jacobi iteration* with damping parameter ω; for $\omega = 1$ it reduces to the ordinary Jacobi iteration.

It can be shown[50] that the spectral radius $\rho(M_{\mathrm{DJ},\omega})$ of the damped Jacobi iteration matrix is less than 1, so that the iteration (2.9) converges exponentially to the solution ϕ. This would appear to be a happy situation. Unfortunately, however, the convergence factor $\rho(M_{\mathrm{DJ},\omega})$ is usually very close to 1, so that many iterations are required in order to reduce the error $\|\phi^{(n)} - \phi\|$ to a small fraction of its initial value. Insight into this phenomenon can be gained by considering the simple *model problem* in which the domain Ω is a square $\{1, \ldots, L\} \times \{1, \ldots, L\}$. In this case we can solve exactly for the eigenvectors and eigenvalues of $M_{\mathrm{DJ},\omega}$: they are

$$\phi_x^{(p)} = \sin p_1 x_1 \sin p_2 x_2 \ , \tag{2.10a}$$

$$\lambda_p = (1-\omega) + \frac{\omega}{2}(\cos p_1 + \cos p_2) \ , \tag{2.10b}$$

where

$$p_1, p_2 = \frac{\pi}{L+1}, \frac{2\pi}{L+1}, \ldots, \frac{L\pi}{L+1} \ .$$

The spectral radius of $M_{\mathrm{DJ},\omega}$ is the eigenvalue of largest magnitude, namely

$$\begin{aligned} \rho(M_{\mathrm{DJ},\omega}) &= \lambda_{\pi/(L+1),\pi/(L+1)} \\ &= 1 - \omega\left[1 - \cos\frac{\pi}{L+1}\right] \\ &= 1 - O(L^{-2}) \ . \end{aligned} \tag{2.11}$$

It follows that $O(L^2)$ iterations are needed for the damped Jacobi iteration to converge adequately. This represents an enormous computational labor when L is large.

It is easy to see what is going on here: *the slow modes* ($\lambda_p \approx 1$) *are the long-wavelength modes* ($p_1, p_2 \ll 1$). [If $\omega \approx 1$, then some modes with wave number $p = (p_1, p_2) \approx (\pi, \pi)$ have eigenvalue $\lambda_p \approx -1$ and so also are slow. This phenomenon can be avoided by taking ω significantly less than 1; for simplicity we shall henceforth take $\omega = \frac{1}{2}$, which makes $\lambda_p \ge 0$ for all p.] It is also easy to see physically *why* the long-wavelength modes are slow. The key fact is that the (damped) Jacobi iteration is *local*: in a single step of the algorithm, "information" is transmitted only to nearest neighbors. One might guess that this "information" executes a random walk around the lattice; and for the true solution to be reached, "information" must propagate from the boundaries to the interior (and back and forth until "equilibrium" is attained). This takes a time of order L^2, in agreement with (2.11). (We remark that this random-walk picture can be made rigorous.[51])

This is an example of a *critical phenomenon*, in precisely the same sense that the term is used in statistical mechanics.[52] The Laplace operator $A = -\Delta$ is critical, inasmuch as its Green function A^{-1} has long-range correlations (power-law decay in dimension $d > 2$, or growth in $d \le 2$). This means that the solution of Poisson's equation in one region of the lattice depends strongly on the solution in distant regions of the lattice; "information" must propagate globally in order for "equilibrium" to be reached. Put another way, excitations at many length scales are significant, from one lattice spacing at the smallest to the entire lattice at the largest. The situation would be very different if we were to consider instead the Helmholtz-Yukawa equation $(-\Delta + m^2)\phi = f$ with $m > 0$: its Green function has exponential decay with characteristic length m^{-1}, so that regions of the lattice separated by distances $\gg m^{-1}$ are essentially decoupled. In this case, "information" need only propagate a distance of order $\min(m^{-1}, L)$ in order for "equilibrium" to be reached. This takes a time of order $\min(m^{-2}, L^2)$, an estimate which can be confirmed rigorously by computing the obvious generalization of (2.10) and (2.11). On the other hand, as $m \to 0$ we recov-

er the Laplace operator with its attendant difficulties: $m=0$ is a *critical point*. We have here an example of *critical slowing down* in classical physics.

B. The multigrid method

The general structure of a remedy should now be obvious to physicists reared on the renormalization group: do not try to deal with all length scales at once, but define instead a *sequence* of problems in which each length scale, beginning with the smallest and working towards the largest, can be dealt with separately. An algorithm of precisely this form was proposed in 1964 by the Soviet numerical analyst Fedorenko,[26] and is now called the *multigrid method.*

Note first that the *only* slow modes in the damped Jacobi iteration are the long-wavelength modes (provided that ω is not near 1): as long as, say $\max(p_1,p_2)\geq\pi/2$, we have $0\leq\lambda_p\leq\frac{3}{4}$ (for $\omega=\frac{1}{2}$), independent of L. It follows that the short-wavelength components of the error $e^{(n)}\equiv\phi^{(n)}-\phi$, and hence also of the residual $r^{(n)}\equiv A\phi^{(n)}-f=Ae^{(n)}$, can be effectively killed by a few (say, five or ten) damped Jacobi iterations. The remaining error (or residual) has primarily long-wavelength components, and so is slowly varying in x space. But a slowly varying function can be well represented on a coarser grid: if, for example, we were told $e_x^{(n)}$ only at *even* values of x, we could nevertheless reconstruct with high accuracy the function $e_x^{(n)}$ at *all* x by, say, linear interpolation. This suggests an improved algorithm for solving (2.1): perform a few damped Jacobi iterations on the original grid, until the error and residual are smooth in x space; then transfer (restrict) the residual to a coarser grid and use it as the right-hand side for a Poisson equation on the coarser grid; perform a few damped Jacobi iterations on the coarser grid; and then transfer (interpolate) the result back to the original (fine) grid and add it to the current approximate solution.

There are two advantages to performing the damped Jacobi iterations on the coarse grid. Firstly, the iterations take less work, because there are fewer lattice points on the coarse grid (2^{-d} times as many for a factor-of-2 coarsening in d dimensions). Secondly, with respect to the coarse grid the long-wavelength modes no longer have such long wavelength: *their wavelength has been halved* (i.e., their wave number has been doubled). This suggests that those modes with, say, $\max(p_1,p_2)\geq\pi/4$ can be effectively killed by a few damped Jacobi iterations on the coarse grid. And then we can transfer the remaining (smooth) residual to a yet coarser grid, and so on recursively. These are the essential ideas of the multigrid method.

Let us now give a precise definition of the general multigrid algorithm. Our goal is to solve the equation $A\phi=f$, where A is a nonsingular linear operator from an N-dimensional real vector space U to another N-dimensional real vector space V. (It clarifies matters both mathematically and physically to distinguish between U and V.) In order to specify the algorithm we must specify the following ingredients:

(1) A sequence of *coarse-grid spaces* $U_M\equiv U, U_{M-1}, U_{M-2}, \ldots, U_0$ and $V_M\equiv V, V_{M-1}, V_{M-2}, \ldots, V_0$. Here $\dim U_l=\dim V_l\equiv N_l$ for $0\leq l\leq M$, and $N=N_M>N_{M-1}>N_{M-2}>\cdots>N_0$.

(2) *Restriction* (or "averaging") *operators* $r_{l-1,l}$: $V_l\to V_{l-1}$ for $1\leq l\leq M$.

(3) *Prolongation* (or "interpolation") *operators* $p_{l,l-1}$: $U_{l-1}\to U_l$ for $1\leq l\leq M$.

(4) *Coarse-grid operators* A_l: $U_l\to V_l$ for $0\leq l\leq M-1$. Of course we take $A_M\equiv A$. Each of the operators A_l is assumed to be nonsingular.

(5) *Basic* (or "smoothing") *iterations* $\mathcal{S}_l$: $U_l\times V_l\to U_l$ for $0\leq l\leq M$. The role of $\mathcal{S}_l$ is to take an approximate solution ϕ_l' to the equation $A_l\phi_l=f_l$ and compute a new (hopefully better) approximate solution $\phi_l''=\mathcal{S}_l(\phi_l',f_l)$. (For the present we can imagine that $\mathcal{S}_l$ consists of a few iterations of damped Jacobi for the operator A_l.) Most generally, we shall use two smoothing iterations, $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$; they may be the same, but need not be.

(6) *Cycle control parameters* (integers) $\gamma_l\geq 1$ for $1\leq l\leq M$, which control the number of times that the coarse grids are visited.

We discuss these ingredients in more detail below.

The multigrid algorithm is then defined recursively[53] as follows:

procedure mgm(l,ϕ,f)

comment This algorithm takes an approximate solution ϕ to the equation $A_l\phi=f$, and overwrites it with a better approximation .

$\phi\leftarrow\mathcal{S}_l^{\text{pre}}(\phi,f)$

if $l>0$ **then**

$d\leftarrow -r_{l-1,l}(A_l\phi-f)$

$\psi\leftarrow 0$

for $j=1$ **until** γ_l **do** mgm$(l-1,\psi,d)$

endif

$\phi\leftarrow\phi+P_{l,l-1}\psi$

$\phi\leftarrow\mathcal{S}_l^{\text{post}}(\phi,f)$

end (2.12)

Here is what is going on: We wish to solve the equation $A_l\phi=f$, and are given as input an approximate solution. The algorithm consists of three steps.

(1) *Presmoothing.* We apply a few iterations of the basic smoother (e.g., damped Jacobi) to the given approximate solution. This produces a better approximate solution in which the high-frequency (short-wavelength) components of the error have been reduced significantly. The low-frequency (long-wavelength) components of the error are, however, still large.

(2) *Coarse-grid correction.* We compute the residual $A_l\phi-f$ and transfer it to the next coarser grid (level $l-1$) using the restriction operator $r_{l-1,l}$. We then use the result d as the right-hand side of the auxiliary equation $A_{l-1}\psi=d$, which we solve approximately by γ_l

iterations of the multigrid algorithm at level $l-1$ (recursive definition) with an initial guess $\psi=0$. We then transfer this approximation solution ψ back to grid l using the prolongation operator $p_{l,l-1}$, and use it to correct our approximate solution ϕ. The goal of this coarse-grid correction is to reduce significantly the low-frequency components of the error in ϕ (hopefully without creating large new high-frequency error components).

(3) *Post smoothing.* We apply, for good measure, a few more iterations of the basic smoother. (This would protect against any high-frequency error components which may inadvertently have been created by the coarse-grid correction step.)

The foregoing constitutes, of course, a single step of the multigrid algorithm. In practice this step would be repeated several times, as in any other iteration, until the error has been reduced to an acceptably small value. The advantage of multigrid over the traditional (e.g., damped Jacobi) iterative methods is that, with a suitable choice of the ingredients $r_{l-1,l}$, $p_{l,l-1}$, A_l and so on, only a few (maybe five or ten) iterations are needed to reduce the error to a small value, *independent of the lattice size L.* This contrasts favorably with the behavior (2.11) of the damped Jacobi method, in which $O(L^2)$ iterations are needed.

For readers who prefer mathematical rather than algorithmic definitions, we give the following equivalent definition of the multigrid iteration: Assume that each smoother $\mathcal{S}_l^{\text{pre}}$ is a stationary linear iteration of the form

$$\mathcal{S}_l^{\text{pre}}(\phi_l, f_1) = S_l^{\text{pre}}\phi_l + T_l^{\text{pre}} f_1 \tag{2.13}$$

with

$$T_l^{\text{pre}} = (I - S_l^{\text{pre}}) A_l^{-1} , \tag{2.14}$$

and analogously for $\mathcal{S}_l^{\text{post}}$. Then the multigrid iteration $\text{mgm}(l,\cdot,\cdot)$ is likewise a stationary linear iteration of the form

$$\phi_l \leftarrow M_l \phi_l + N_l f_l \tag{2.15}$$

with

$$N_l = (I - M_l) A_l^{-1} , \tag{2.16}$$

and the multigrid iteration matrices M_l are defined inductively as follows:

$$M_0 = S_0^{\text{post}} S_0^{\text{pre}} , \tag{2.17a}$$

$$M_l = S_l^{\text{post}} [I - p_{l,l-1}(I - M_{l-1}^{\gamma_l}) A_{l-1}^{-1} r_{l-1,l} A_l] S_l^{\text{pre}}$$

$$\text{for } l \geq 1 . \tag{2.17b}$$

The multigrid algorithm is thus a general framework; the user has considerable freedom in choosing the specific ingredients, which must be adapted to the specific problem. We now discuss briefly each of these ingredients; more details can be found in Chap. 3 of the book of Hackbusch.[30]

Coarse grids. Most commonly one uses a uniform factor-of-2 coarsening between each grid Ω_l and the next coarser grid Ω_{l-1}. The coarse-grid points could be either a subset of the fine-grid points [Fig. 1(a)] or a subset of the dual lattice [Fig. 1(b)]. These schemes have obvious generalizations to higher-dimensional cubic lattices. In dimension $d=2$, another possibility is a uniform factor-of-$\sqrt{2}$ coarsening [Fig. 1(c)]; note that the coarse grid is again a square lattice, rotated by 45°. Figures 1(a)–1(c) are often referred to as "standard coarsening," "staggered coarsening," and "red-black (or checkerboard) coarsening," respectively. Coarsenings by a larger factor (e.g., 3) could also be considered, but are generally disadvantageous. Note that each of the above schemes works also for periodic boundary conditions provided that the linear size L_l of the grid Ω_l is *even*. For this reason it is most convenient to take the linear size $L \equiv L_M$ of the original (finest) grid $\Omega \equiv \Omega_M$ to be a power of 2, or at least a power of 2 times a small integer. Other definitions of coarse grids (e.g., anisotropic coarsening) could be appropriate in specific problems.[54]

Restriction operators. If the coarse-grid points are a subset of the fine-grid points, as in Figs. 1(a) and 1(c), then one simple restriction operator is *trivial restriction*: just define

$$(r_{l-1,l}\phi_l)_x = (\phi_l)_x \quad \text{for all } x \in \Omega_{l-1} \subset \Omega_l , \tag{2.18}$$

However, this restriction is in most cases too crude.[55] A better restriction operator would incorporate some degree of local averaging ("coarse graining"), in order to accentuate the low-frequency components. For a coarse grid as in Fig. 1(b), a natural choice is *block averaging*:

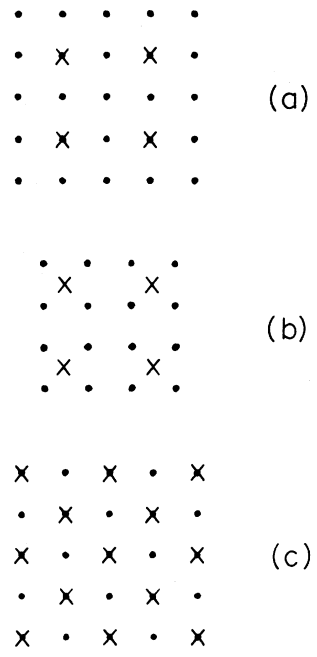

FIG. 1. Possible choices of the coarse grid, shown here in dimension $d=2$. Dots are fine-grid sites; crosses are coarse-grid sites. (a) Standard coarsening. (b) Staggered coarsening. (c) Red-black (checkerboard) coarsening.

$$(r_{l-1,l}\phi_l)_x = \tfrac{1}{4}[(\phi_l)_{x_1+1/2,x_2+1/2}+(\phi_l)_{x_1-1/2,x_2+1/2} +(\phi_l)_{x_1+1/2,x_2-1/2}+(\phi_l)_{x_1-1/2,x_2-1/2}] \tag{2.19}$$

(illustrated here for $d=2$). This restriction can be represented in an obvious shorthand notation by the stencil

$$\begin{bmatrix} \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} \end{bmatrix} . \tag{2.20}$$

For a coarse grid as in Fig. 1(a), one natural choice is the *nine-point restriction* (also called "full weighting")

$$\begin{bmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{bmatrix} . \tag{2.21}$$

All these restriction operators can easily be generalized to higher dimensions.

Prolongation operators. If the coarse-grid points are a subset of the fine-grid points, as in Figs. 1(a) and 1(c), then one simple prolongation operator is *trivial prolongation*: just define

$$(p_{l,l-1}\phi_{l-1})_x = \begin{cases} (\phi_{l-1})_x & \text{if } x \in \Omega_{l-1} , \\ 0 & \text{otherwise} . \end{cases} \tag{2.22}$$

However, this prolongation is in most cases too crude. A better prolongation operator would incorporate some degree of local smoothing, in order to accentuate the low-frequency components. For a coarse grid as in Fig. 1(b), a natural choice is *piecewise-constant injection*:

$$(p_{l,l-1}\phi_{l-1})_{x_1\pm 1/2,x_2\pm 1/2} = (\phi_l)_{x_1,x_2} \quad \text{for all } x \in \Omega_{l-1} . \tag{2.23}$$

It can be represented by the stencil

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} . \tag{2.24}$$

For a coarse grid as in Fig. 1(a), a natural choice is *piecewise-linear interpolation*, one example of which is the *nine-point prolongation*

$$\begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix} . \tag{2.25}$$

Higher-order interpolations (e.g., quadratic or cubic) can also be considered. All these prolongation operators can easily be generalized to higher dimensions.

Given a prolongation operator, one can always define a restriction operator to be its adjoint (transpose), and vice versa. For example, the adjoint of trivial prolongation is trivial restriction, the adjoint of piecewise-constant injection is block averaging, and the adjoint of the nine-point prolongation is the nine-point restriction.

We have ignored here some important subtleties concerning the treatment of the boundaries in defining the restriction and prolongation operators. Fortunately we shall not have to worry much about this problem, since most applications in quantum field theory use periodic boundary conditions.

Coarse-grid operators. The operator A_l is supposed to model roughly the behavior of the original operator $A = A_M$ when acting on "smooth" functions (i.e., those of wavelength $\gtrsim 2^{M-l}$). One very natural choice of the A_l is given by the *Galerkin definition*

$$A_{l-1} = r_{l-1,l} A_l p_{l,l-1} . \tag{2.26}$$

Other possibilities could also be considered, but we shall stick to Galerkin. Note that if A is the nearest-neighbor Laplacian, and block averaging and piecewise-constant interpolation are used, then the Galerkin coarse-grid operators A_l are again nearest-neighbor Laplacians (suitably normalized). On the other hand, if higher-order interpolations are used, then the Galerkin coarse-grid operators usually involve next-nearest-neighbor couplings as well.[56]

Smoothing iterations. We have already discussed the damped Jacobi iteration as one possible smoother. Note that in this method only the "old" values $\phi^{(n)}$ are used on the right-hand side of (2.8) and (2.9), even though for some of the terms the "new" value $\phi^{(n+1)}$ may already have been computed. An alternative algorithm is to use at each stage on the right-hand side the "newest" available value. This algorithm is called the *Gauss-Seidel iteration.*[57] Note that the Gauss-Seidel algorithm, unlike the Jacobi algorithm, depends on the *ordering* of the grid points. For example, if a two-dimensional grid is swept in *lexicographic order*, $(1,1),(2,1),\ldots,(L,1),(1,2),(2,2),\ldots,(L,2),\ldots,(1,L),(2,L),\ldots,(L,L)$, then the Gauss-Seidel iteration becomes

$$\phi^{(n+1)}_{x_1,x_2} = \tfrac{1}{4}(\phi^{(n)}_{x_1+1,x_2}+\phi^{(n+1)}_{x_1-1,x_2}+\phi^{(n)}_{x_1,x_2+1} +\phi^{(n+1)}_{x_1,x_2-1}+f_{x_1,x_2}) . \tag{2.27}$$

Another convenient ordering is the *red-black* (or *checkerboard*) *ordering*, in which the "red" sublattice $\Omega^r = \{x \in \Omega : x_1 + \cdots + x_d \text{ is even}\}$ is swept first, followed by the "black" sublattice $\Omega^b = \{x \in \Omega : x_1 + \cdots + x_d \text{ is odd}\}$. Note that the ordering of the grid points *within* each sublattice is irrelevant [for the usual nearest-neighbor Laplacian (2.1)], since the matrix A does not couple sites of the same color. This means that red-black Gauss-Seidel is particularly well suited to vector or parallel computation. Note that the red-black ordering makes sense with periodic boundary conditions only if the linear size L_l of the grid Ω_l is even.

The Gauss-Seidel iteration, like the Jacobi iteration, can be generalized by introducing a parameter ω: for example, in the two-dimensional lexicographic case we would have

$$\phi^{(n+1)}_{x_1,x_2} = (1-\omega)\phi^{(n)}_{x_1,x_2} + \frac{\omega}{4}(\phi^{(n)}_{x_1+1,x_2} + \phi^{(n+1)}_{x_1-1,x_2} + \phi^{(n)}_{x_1,x_2+1} + \phi^{(n+1)}_{x_1,x_2-1} + f_{x_1,x_2}) \ . \tag{2.28}$$

It turns out[58] that this iteration converges if and only if $0<\omega<2$. This algorithm is called the *successive overrelaxation (SOR) iteration* with relaxation parameter ω. (Of course, if $\omega<1$ it really ought to be called "under relaxation.") For $\omega=1$ it reduces to the Gauss-Seidel iteration.

As a stand-alone iterative solver, Gauss-Seidel can be shown[59] to converge roughly twice as fast as Jacobi (it also requires half as much storage). In particular, the underlying $O(L^2)$ critical slowing-down is unchanged. For SOR, on the other hand, the situation is rather different: the optimal relaxation parameter ω_{opt} is $2-O(L^{-1})$, and if this value of ω is used then the critical slowing down is only $O(L)$—far better than for Jacobi or Gauss-Seidel, but still quite severe. (Also, in practical problems ω_{opt} may not be known.)

For a smoothing iteration within a multigrid algorithm, however, the considerations are quite different. The performance of the smoother on the *low-frequency* modes (which determines its critical slowing down as a stand-alone solver) is irrelevant, since the low-frequency modes are dealt with by the coarse-grid correction step. Rather, it is the rate at which the *high-frequency* error components are reduced—the so-called *smoothing rate*—that is important. It turns out[60] that the optimal value of ω for SOR smoothing is 1, i.e., the ordinary Gauss-Seidel iteration. Gauss-Seidel is also a better smoother than damped Jacobi with its optimal ω.

Many other smoothing iterations can be considered,[61] and can be advantageous in anisotropic or otherwise singular problems. But we shall stick to ordinary Gauss-Seidel, usually with red-black ordering.

Thus, $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$ will consist, respectively, of m_1 and m_2 iterations of the Gauss-Seidel algorithm. The balance between presmoothing and postsmoothing is usually not very crucial; only the total m_1+m_2 seems to matter much. Indeed, one (but not both) of m_1 or m_2 could be zero, i.e., either the presmoothing or the postsmoothing could be omitted entirely. Increasing m_1 and m_2 improves the convergence rate of the multigrid iteration, but at the expense of increased computational labor per iteration. The optimal tradeoff seems to be achieved in most cases with m_1+m_2 between about 2 and 4 (Ref. 62). The coarsest grid Ω_0 is a special case: it usually has so few grid points (perhaps only one) that $\mathcal{S}_0$ can be an exact solver.

Cycle control parameters. Usually the parameters γ_l are all taken to be equal, i.e., $\gamma_l=\gamma\geq 1$ for $1\leq l\leq M$. Then one iteration of the multigrid algorithm at level M comprises one visit to grid M, γ visits to grid $M-1$, γ^2 visits to grid $M-2$, and so on. Thus, γ determines the degree of emphasis placed on the coarse-grid updates. ($\gamma=0$ would correspond to the pure Gauss-Seidel iteration on the finest grid alone.)

We can now estimate the computational labor required for one iteration of the multigrid algorithm. Each visit to a given grid involves m_1+m_2 Gauss-Seidel sweeps on that grid, plus some computation of the residual, the restriction, and the prolongation. The work involved is proportional to the number of lattice points on that grid. Let W_l be the work required for these operations on grid l. Then, for grids defined by a factor-of-2 coarsening in d dimensions, we have

$$W_l \approx 2^{-d(M-l)} W_M \ , \tag{2.29}$$

so that the total work for one multigrid iteration is

$$\begin{aligned} \text{work(MG)} &= \sum_{l=M}^{0} \gamma^{M-l} W_l \\ &\approx W_M \sum_{l=M}^{0} (\gamma 2^{-d})^{M-l} \\ &\leq W_M (1-\gamma 2^{-d})^{-1} \quad \text{if } \gamma < 2^d \ . \end{aligned} \tag{2.30}$$

Thus, provided that $\gamma<2^d$, the work required for one entire multigrid iteration is no more than $(1-\gamma 2^{-d})^{-1}$ times the work required for m_1+m_2 Gauss-Seidel iterations (plus a little auxiliary computation) on the finest grid alone—*irrespective of the total number of levels.* The most common choices are $\gamma=1$ (which is called the V cycle) and $\gamma=2$ (the W cycle).

For certain classes of operators A and suitable choices of the coarse grids, restrictions, prolongations, coarse-grid operators, smoothing iterations, and cycle control parameters, it can be proven rigorously[63] that the multigrid iteration matrices M_l satisfy a *uniform* bound

$$\|M_l\| \leq C < 1 \ , \tag{2.31}$$

valid *irrespective of the total number of levels.* Thus, a fixed number of multigrid iterations (maybe five or ten) are sufficient to reduce the error to a small value, *independent of the lattice size L.* In other words, *critical slowing down has been completely eliminated.*

The rigorous convergence proofs are somewhat arcane, so we cannot describe them here in any detail,[63] but certain general features are worth noting. The convergence proofs are most straightforward when linear or higher-order interpolation and restriction are used, and $\gamma>1$ (e.g., the W cycle). When either low-order interpolation (e.g., piecewise constant) or $\gamma=1$ (the V cycle) is used, the convergence proofs become much more delicate. Indeed, if *both* piecewise-constant interpolation and a V cycle are used, then the uniform bound (2.31) has *not* yet been proven, and it is most likely *false.* To some extent these features may be artifacts of the current methods of proof, but we suspect that they do also reflect real properties of the multigrid method, and so the convergence proofs may serve as guidance for practice. For example, in our work we have used piecewise-constant interpolation (so as to preserve the simple nearest-neighbor coupling on the coarse grids), and thus for safety we stick to the W cycle. There is in any case much room for further research, both theoretical and experimental.

To recapitulate, the extraordinary efficiency of the

multigrid method arises from the combination of two key features.

(1) The convergence estimate (2.31). This means that only $O(1)$ iterations are needed, independent of the lattice size L.

(2) The work estimate (2.30). This means that each iteration requires only a computational labor of order L^d (the fine-grid lattice volume).

It follows that the complete solution of the linear system $A\phi = f$ requires a computational labor of order L^d.

C. Variational approach

We now turn to an important special case of the multigrid algorithm. Suppose that the space V can be identified with U^* (the dual space of U) in such a way that the operator A is *symmetric*,

$$\langle \phi, A\phi' \rangle = \langle A\phi, \phi' \rangle \quad \text{for all } \phi, \phi' \in U \ , \tag{2.32}$$

and *positive-definite*,

$$\langle \phi, A\phi \rangle > 0 \quad \text{for all } \phi \in U,\ \phi \neq 0 \ . \tag{2.33}$$

The problem of solving the linear equation $A\phi = f$ is then equivalent to a variational problem, that of minimizing the quadratic function

$$H(\phi) = \tfrac{1}{2}\langle \phi, A\phi \rangle - \langle f, \phi \rangle \ , \tag{2.34}$$

and the variational point of view often leads to useful insights.[64]

For example, let us look at the coarse-grid-correction phase of the multigrid algorithm from the variational point of view. In this phase, we replace the current approximate solution ϕ by a hopefully better approximate solution $\phi + p_{l,l-1}\psi$, where ψ lies in the coarse-grid subspace U_{l-1}. A sensible goal is to attempt to choose ψ so as to minimize H; that is, we attempt to minimize

$$H_{l-1}(\psi) \equiv H_l(\phi + p_{l,l-1}\psi) \tag{2.35a}$$

$$= \tfrac{1}{2}\langle \psi, A_{l-1}\psi \rangle - \langle d, \psi \rangle + \text{const} \ , \tag{2.35b}$$

where

$$A_{l-1} \equiv p^*_{l,l-1} A_l p_{l,l-1} \ , \tag{2.36a}$$

$$d \equiv p^*_{l,l-1}(f - A_l\phi) \ . \tag{2.36b}$$

This means that we attempt to solve the equation $A_{l-1}\psi = d$, where A_{l-1} and d are as defined above. Comparison with (2.12) reveals that this is precisely what the multigrid algorithm does, in the special case with the following properties.

(1) Each restriction operator is the adjoint of the corresponding prolongation, i.e., $r_{l-1,l} = p^*_{l,l-1}$.

(2) The coarse-grid operators are given by the Galerkin definition $A_{l-1} = r_{l-1,l} A_l p_{l,l-1}$.

We call these the *variational choices* of restriction and coarse-grid operators; they are completely determined by the original operator A and the prolongation operators $p_{l,l-1}$. The variational point of view shows why these choices are so natural. [Indeed, in the corresponding Monte Carlo problem (Sec. VII) the variational choices are *mandatory*.]

If we assume further (without any loss of generality) that the prolongation operators $p_{l,l-1}$ have maximal rank (i.e., zero nullspace), then the multigrid setup takes on a very elegant Hilbert-space structure.

(3) The fine-grid space $U \equiv U_M$ can be given the structure of a Hilbert space with inner product

$$(\phi, \phi')_A \equiv \langle \phi, A\phi' \rangle \ . \tag{2.37}$$

The corresponding norm

$$\|\phi\|_A \equiv (\phi, \phi)_A^{1/2} = \langle \phi, A\phi \rangle^{1/2} \tag{2.38}$$

is called the *energy norm*.

(4) Each coarse-grid space U_l can be identified with a linear subspace of U, namely, its image under the prolongation map $p_{M,l} \equiv p_{M,M-1} p_{M-1,M-2} \cdots p_{l+1,l}$. These spaces are nested: $U_0 \subset U_1 \subset \cdots \subset U_M \equiv U$.

(5) Each restriction map

$$r_{l,m} \equiv r_{l,l+1} r_{l+1,l+2} \cdots r_{m-1,m} \quad (l < m \leq M)$$

is precisely the $(\ ,\ )_A$-orthogonal projection of U_m onto U_l.

(6) The quadratic form $\langle \cdot, A_l \cdot \rangle$ induced on U_l by the coarse-grid operator A_l is precisely the restriction of the quadratic form $\langle \cdot, A_l \cdot \rangle$ on U to the subspace U_l. In particular, A_l is symmetric and positive definite (hence nonsingular).

The multigrid algorithm is thus seen as a sequence of approximate minimizations of H in the subspaces $U_M, U_{M-1}, \ldots, U_0$. The coarse-grid subspaces $U_{M-1}, \ldots, U_0$ are singled out for special treatment because they are the subspaces in which the Gauss-Seidel approximate minimization is least effective.

The variational point of view also gives insight into the Gauss-Seidel algorithm and its relation with the multigrid method. One natural class of iterative algorithms for minimizing H are the so-called *directional methods*: let $p_0, p_1, \ldots$ be a sequence of "direction vectors" in U, and define $\phi^{(n+1)}$ to be that vector of the form $\phi^{(n)} + \lambda p_n$ which minimizes H. This is a one-dimensional minimization problem, and an easy computation yields

$$\lambda = -\frac{\langle p_n, A\phi^{(n)} - f \rangle}{\langle p_n, Ap_n \rangle} \ . \tag{2.39}$$

The algorithm thus travels "downhill" from $\phi^{(n)}$ along the line $\phi^{(n)} + \lambda p_n$ until reaching the minimum of H, then switches to direction p_{n+1} starting from this new point $\phi^{(n+1)}$, and so on. For a suitable choice of the direction vectors $p_0, p_1, \ldots$, this method converges[65] to the global minimum of H, which is $\phi = A^{-1}f$.

Now, some iterative algorithms for solving the linear equation $A\phi = f$ can be recognized as special cases of the directional method. For example, the Gauss-Seidel iteration is a directional method in which the direction vectors are chosen to be unit vectors $e_1, e_2, \ldots, e_N$ (i.e., vectors which take the value 1 at a single grid point and zero

at all others), where $N = \dim U$. (One step of the Gauss-Seidel iteration corresponds to N steps of the directional method.) Similarly, it is not hard to see[66] that the multigrid iteration with the variational choices of restriction and coarse-grid operators, and with Gauss-Seidel smoothing at each level, is itself a directional method: some of the direction vectors are the unit vectors $e_1^{(M)}, e_2^{(M)}, \ldots, e_{N_M}^{(M)}$ of the fine-grid space, but other direction vector are the images in the fine-grid space of the unit vectors of the coarse-grid spaces, i.e., they are $p_{M,l}e_1^{(l)}, p_{M,l}e_2^{(l)}, \ldots, p_{M,l}e_{N_l}^{(l)}$. The exact order in which these direction vectors are interleaved depends on the parameters m_1, m_2, and γ which define the cycling structure of the multigrid algorithm. For example, if $m_1 = 1$, $m_2 = 0$, and $\gamma = 1$, the order of the direction vectors is $\{M\}, \{M-1\}, \ldots, \{0\}$ where $\{l\}$ denotes the sequence $p_{M,l}e_1^{(l)}, p_{M,l}e_2^{(l)}, \ldots, p_{M,l}e_{N_l}^{(l)}$. If $m_1 = 0$, $m_2 = 1$, and $\gamma = 1$, the order is $\{0\}, \{1\}, \ldots, \{M\}$. The reader is invited to work out other cases.

Thus, the multigrid algorithm under the variational conditions (1) and (2) is a directional method in which the direction vectors include both "single-site modes" $\{M\}$ and also "collective modes" $\{M-1\}, \{M-2\}, \ldots, \{0\}$ on all length scales. For example, if $p_{l,l-1}$ is piecewise-constant injection, then the direction vectors are characteristic functions χ_B (i.e., functions which are 1 on the block $B \subset \Omega$ and zero outside B), where the sets B are successively single sites, cubes of side 2, cubes of side 4, and so on. Similarly, if $p_{l,l-1}$ is linear interpolation, then the direction vectors are triangular waves of various widths.

The multigrid algorithm has thus an alternative interpretation as a collective-mode algorithm working solely in the fine-grid space U. We emphasize that this "unigrid" viewpoint[66] is mathematically fully equivalent to the recursive definition (2.12). But it gives, we think, an important additional insight into what the multigrid algorithm is really doing.

For example, for the simple model problem (Poisson equation in a square), we know that the "correct" collective modes are sine waves, in the sense that these modes diagonalize the Laplacian, so that in this basis the Jacobi or Gauss-Seidel algorithm would give the exact solution in a *single* iteration ($M_{\text{Jacobi}} = M_{\text{GS}} = 0$). On the other hand, the multigrid method uses square-wave (or triangular-wave) updates, which are not exactly the "correct" collective modes. Nevertheless, the multigrid convergence proofs[63] assure us that they are "close enough": the norm of the multigrid iteration matrix M_l is bounded away from 1, uniformly in the lattice size, so that an accurate solution is reached in a *very few* MG iterations (in particular, critical slowing-down is completely eliminated). This viewpoint also explains why MG convergence is more delicate for piecewise-constant interpolation than for piecewise linear: the point is that a sine wave (or other slowly varying function) can be approximated to arbitrary accuracy (in energy norm) by piecewise-linear functions but *not* by piecewise-constant functions.[67]

We remark that McCormick and Ruge[66] have advocated the "unigrid" (UG) idea not just as an alternate point of view on the multigrid algorithm, but as an alternate *computational procedure*. To be sure, the unigrid method is somewhat simpler to program, and this could have pedagogical advantages. But one of the key properties of the multigrid method, namely the $O(L^d)$ computational labor per iteration, is sacrificed in the unigrid scheme. Instead of (2.29)–(2.30) one has

$$W_l \approx W_M \tag{2.40}$$

and hence

$$\begin{aligned} \text{work(UG)} &\approx W_M \sum_{l=M}^{0} \gamma^{M-l} \\ &\sim \begin{cases} MW_M & \text{if } \gamma = 1 , \\ \gamma^M W_M & \text{if } \gamma > 1 . \end{cases} \end{aligned} \tag{2.41}$$

Since $M \approx \log_2 L$ and $W_M \sim L^d$, we obtain

$$\text{work(UG)} \sim \begin{cases} L^d \log L & \text{if } \gamma = 1 , \\ L^{d+\log_2 \gamma} & \text{if } \gamma > 1 . \end{cases} \tag{2.42}$$

For a V cycle the additional factor of $\log L$ is perhaps not terribly harmful, but for a W cycle the additional factor of L is a severe drawback [though not as severe as the $O(L^2)$ critical slowing down of the traditional algorithms]. Thus, we do *not* advocate the use of unigrid as a computational method if there is a viable multigrid alternative. The unigrid method could, however, be of interest in cases where a true multigrid algorithm is unfeasible, as may occur for non-Abelian lattice gauge theories (Sec. V).

III. DETERMINISTIC MULTIGRID METHOD: NONLINEAR SYSTEMS, LINEAR STATE SPACE (ADDITIVE MG)

In this section we extend the deterministic multigrid method to *nonlinear* systems of equations on a *linear* state space; as in the preceding section, the coarse-grid-correction updates are *additive*. We restrict attention to problems posed in variational form: our goal is thus to find the absolute minimum of a given nonquadratic Hamiltonian. We do not purport to treat all variants of the nonlinear multigrid method, or even necessarily the best ones; nor do we make any serious attempt to evaluate the performance of the algorithms we discuss. Indeed, the performance of nonlinear multigrid methods is highly model dependent, and is not well understood at present.[68] Rather, our primary goal in this section is to describe the structure of a multigrid algorithm for an arbitrary (not necessarily quadratic) Hamiltonian, and in particular the definition of the coarse-grid Hamiltonian; this structure will be identical in the corresponding stochastic (multigrid Monte Carlo) algorithm.

The setup is as follows: On an N-dimensional real vector space U, we are given a real-valued function ("Hamiltonian") H, which we assume (for simplicity) to have a unique absolute minimum; our goal is to find this minimum. In order to specify the multigrid algorithm we must specify the following ingredients.

(1) A sequence of *coarse-grid configuration spaces* $U_M \equiv U, U_{M-1}, U_{M-2}, \ldots, U_0$. Here $\dim U_l \equiv N_l$ for $0 \le l \le M$, and $N = N_M > N_{M-1} > N_{M-2} > \cdots > N_0$.

(2) *Prolongation* (or "interpolation") *operators* $p_{l,l-1}$: $U_{l-1} \to U_l$ for $1 \le l \le M$.

(3) *Basic* (or "smoothing") *iterations* $\mathcal{S}_l$: $U_l \times \mathcal{H}_l \to U_l$ for $0 \le l \le M$. Here $\mathcal{H}_l$ is a space of "possible Hamiltonians" defined on U_l; we discuss this in more detail below. The role of $\mathcal{S}_l$ is to take an approximate minimizer ϕ_l' of the Hamiltonian H_l and compute a new (hopefully better) approximate minimizer $\phi_l'' = S_l(\phi_l', H_l)$. Most generally, we shall use two smoothing iterations, $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$; they may be the same, but need not be.

(4) *Cycle control parameters* (integers) $\gamma_l \ge 1$ for $1 \le l \le M$, which control the number of times that the coarse grids are visited.

The nonlinear multigrid algorithm is then defined recursively as follows:

procedure $nlmgm(l, \phi, H_l)$

comment This algorithm takes an approximate minimizer ϕ of the Hamiltonian H_l, and overwrites it with a better approximate minimizer .

$\phi \leftarrow \mathcal{S}_l^{\text{pre}}(\phi, H_l)$

if $l > 0$ **then**

compute $H_{l-1}(\cdot) \equiv H_l(\phi + p_{l,l-1}\cdot)$

$\psi \leftarrow 0$

for $j = 1$ **until** γ_l **do** $nlmgm(l-1, \psi, H_{l-1})$

$\phi \leftarrow \phi + p_{l,l-1}\psi$

endif

$\phi \leftarrow \mathcal{S}_l^{\text{post}}(\phi, H_l)$

end (3.1)

The basic idea of this algorithm is the same as that of the linear multigrid algorithm explained in Sec. II. Only two differences require comment: the meaning of the step "*compute* H_{l-1}," and the nature of the nonlinear smoothing iterations $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$.

Computation of H_{l-1}. This is best explained by a concrete example. Suppose that the Hamiltonian H_l on level l is a ϕ^4 theory with nearest-neighbor gradient term and possibly site-dependent coefficients:

$$H_l(\phi) = \frac{\alpha}{2} \sum_{|x-x'|=1} (\phi_x - \phi_{x'})^2 + \sum_x V_x(\phi_x) , \tag{3.2a}$$

where

$$V_x(\phi_x) = \lambda \phi_x^4 + \kappa_x \phi_x^3 + A_x \phi_x^2 + h_x \phi_x . \tag{3.2b}$$

Suppose, further, that the prolongation operator $p_{l,l-1}$ is piecewise-constant injection (2.23). Then the coarse-grid Hamiltonian

$$H_{l-1}(\psi) \equiv H_l(\phi + p_{l,l-1}\psi) \tag{3.3}$$

can easily be computed: it is

$$H_{l-1}(\psi) = \frac{\alpha'}{2} \sum_{|y-y'|=1} (\psi_y - \psi_{y'})^2 + \sum_y V_y'(\psi_y) + \text{const} , \tag{3.4a}$$

where

$$V_y'(\psi_y) = \lambda' \psi_y^4 + \kappa_y' \psi_y^3 + A_y' \psi_y^2 + h_y' \psi_y \tag{3.4b}$$

and

$$\begin{aligned} \alpha' &= 2^{d-1}\alpha , \\ \lambda' &= 2^d \lambda , \\ \kappa_y' &= \sum_{x \in B_y} (4\lambda \phi_x + \kappa_x) , \\ A_y' &= \sum_{x \in B_y} (6\lambda \phi_x^2 + 3\kappa_x \phi_x + A_x) , \\ h_y' &= \sum_{x \in B_y} (4\lambda \phi_x^3 + 3\kappa_x \phi_x^2 + 2A_x \phi_x + h_x) . \end{aligned} \tag{3.4c}$$

Here B_y is the block consisting of those 2^d sites of grid Ω_l which are affected by interpolation from the coarse-grid site $y \in \Omega_{l-1}$ [see Fig. 1(b)]. Note that the coarse-grid Hamiltonian H_{l-1} has the same functional form as the "fine-grid" Hamiltonian H_l: it is specified by the coefficients α', λ', $\{\kappa_y'\}$, $\{A_y'\}$, and $\{h_y'\}$. The step "*compute* H_{l-1}" therefore means to compute these coefficients. Note also the importance of allowing in (3.2) for ϕ^3 and ϕ terms and for site-dependent coefficients: even if these are not present in the original Hamiltonian $H \equiv H_M$, they will be generated on coarser grids.[69] Finally, we emphasize that the coarse-grid Hamiltonian H_{l-1} depends implicitly on the current value of the fine-lattice field $\phi \in U_l$; although our notation suppresses this dependence, it should be kept in mind.

The general strategy should now be clear: We choose classes of Hamiltonians $\mathcal{H}_l$ with the property that if $H_l \in \mathcal{H}_l$ and $\phi \in U_l$, then the coarse-grid Hamiltonian H_{l-1} defined by (3.3) necessarily lies in $\mathcal{H}_{l-1}$. In particular, it is convenient (though not in principle necessary) to choose all the Hamiltonians to have the same "functional form"; this functional form must be one which is stable under the coarsening operation (3.3). For example, $P(\phi)$ theories (i.e., V a polynomial of degree $2n$) can be treated by an obvious generalization of the ϕ^4 example, as can theories with exponential interaction

$$V_x(\phi_x) = c_x^{(1)} e^{\sigma_1 \phi_x} + c_x^{(2)} e^{\sigma_2 \phi_x} + \cdots + c_x^{(n)} e^{\sigma_n \phi_x} \tag{3.5}$$

with given $\sigma_1, \sigma_2, \ldots, \sigma_n$. Note, however, that the simplicity of the coarse-grid Hamiltonian is linked to our use of piecewise-constant (or trivial) injection: the key fact is that each fine-grid site is affected by only *one* coarse-grid site, so the nonlinearities remain strictly local. If we were to use piecewise-linear injection, then the local ϕ_x^4 term in H_l would induce nonlocal (nearest-neighbor and possibly next-nearest-neighbor) nonlinear terms like $\phi_y^2 \phi_{y'}^2$, and

$\phi_y^3\phi_{y'}$ in H_{l-1}. This process of generation of additional interactions does eventually stop, so appropriate classes of Hamiltonians $\mathcal{H}_l$, stable under coarsening, can be found; but they are significantly more complicated than the original ϕ^4 theory. It is for this reason that we have generally preferred to use piecewise-constant injection in our work with nonlinear multigrid methods.

Finally, we remind the reader that in the quadratic case

$$H_l(\phi)=\tfrac{1}{2}\langle \phi, A_l\phi\rangle - \langle f,\phi\rangle \ , \tag{3.6}$$

the nonlinear multigrid algorithm (3.1) is identical to the linear multigrid algorithm (2.12) with the variational definition (2.36a) of the coarse-grid operators (compare with the discussion in Sec. II C).

Smoothing iterations. Numerous algorithms for iterative minimization of a nonlinear function can be found in the numerical-analysis literature:[70] examples are Newton methods, nonlinear Gauss-Seidel methods, conjugate-gradient methods, and combinations of these. Any one of these algorithms could be used for the smoothing iterations $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$. For expository purposes, however, it is convenient to focus on one particular method, nonlinear Gauss-Seidel with exact minimization[71] (NLGSEM). In this algorithm, the grid points are swept in some order (e.g., lexicographic or red-black), and at each stage the Hamiltonian H_l is minimized as a function of a single variable ϕ_x, with all other variables $\{\phi_{x'}\}_{x'\neq x}$ being held fixed. (If the absolute minimizer is nonunique, then one such minimizer is chosen by some arbitrary rule.) This definition of the algorithm presupposes, of course, that it is feasible to carry out the requisite exact one-dimensional minimizations. For example, for a ϕ^4 theory it would be necessary to compute the absolute minimum of a quartic polynomial in one variable. In practice these one dimensional minimizations might themselves be carried out iteratively, e.g., by some variant of Newton's method. We note that for a quadratic Hamiltonian H_l, the nonlinear Gauss-Seidel algorithm with exact minimization reduces to the ordinary Gauss-Seidel algorithm.

Nonlinear Gauss-Seidel with exact minimization is an example of a *directional method* as defined in Sec. II C; the direction vectors are the unit vectors $e_1, e_2, \ldots, e_N$, just as in the ordinary Gauss-Seidel algorithm. Likewise, the nonlinear multigrid algorithm (3.1) is also a directional method, if NLGSEM is used for $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$; the direction vectors are the same as in the corresponding linear multigrid algorithm (see Sec. II C).

The *performance* of the nonlinear multigrid algorithm (3.1) is highly dependent on the details of the Hamiltonian H (and perhaps also on the prolongation operators and smoothing iterations), so we can make only a few general remarks. If the Hamiltonian H is *strictly convex*, then the nonlinear multigrid algorithm (with NLGSEM as the smoothing iteration) is guaranteed to converge to the (unique) absolute minimum ϕ^* of H, irrespective of the initial condition $\phi^{(0)}$; indeed, this is a general result which holds for a wide class of directional methods.[65] Moreover, the asymptotic rate of convergence is determined by the behavior of the corresponding *linear* multigrid algorithm on the quadratic Hamiltonian

$$H_{\text{lin}}(\phi)=\tfrac{1}{2}\langle \phi, H''(\phi^*)\phi\rangle \ . \tag{3.7}$$

(This is because, as $n\to\infty$, the iterates $\phi^{(n)}$ lie in an arbitrarily small neighborhood of ϕ^*, so for asymptotic purposes H can be replaced by its quadratic approximation.[72]) On the other hand, for *nonconvex* Hamiltonians, such as the ϕ^4 theory in the double-well regime, we are unable to say anything definitive. Traditional iterative algorithms can get stuck in local minima of H, and the same can occur in nonlinear multigrid methods. Fortunately, such difficulties should be less severe in the corresponding stochastic algorithm (Sec. VII).

IV. DETERMINISTIC MULTIGRID METHODS: NONLINEAR σ MODELS (MULTIPLICATIVE MG)

In this section we extend the deterministic multigrid method to the problem of minimizing a Hamiltonian of the form arising in nonlinear σ models. The basic multigrid algorithm is the same as in Sec. III; however, the fact that the state space is a *nonlinear manifold* (rather than a vector space) necessitates a different definition of the interpolation operators. In particular, the coarse-grid-correction updates are *multiplicative* rather than additive. Our primary goal in this section is, therefore, to introduce the interpolation operators and coarse-grid Hamiltonians, in preparation for the corresponding stochastic (multigrid Monte Carlo) algorithm. The reader who is anxious to get to the Monte Carlo case [albeit only for free fields and $P(\phi)$ models] can skip this section on a first reading.

Let us also remark that the deterministic multigrid algorithm presented here has direct application to the problem of Landau gauge fixing.[73]

The simplest example is the plane-rotator (XY) model. At each site x there is an angular variable θ_x; angles which are equal modulo 2π are identified. The state space is therefore an $|\Omega|$-dimensional torus, where $|\Omega|$ is the number of sites in the lattice. The Hamiltonian is assumed to be a nearest-neighbor interaction, but we allow the coefficients to be space dependent:

$$H_l(\theta)=\sum_{|x-x'|=1}\left[\alpha_{xx'}\cos(\theta_x-\theta_{x'})+\beta_{xx'}\sin(\theta_x-\theta_{x'})\right] . \tag{4.1}$$

Of course, the case of primary physical interest is the usual translation-invariant nonfrustrated XY model, $\alpha_{xx'}=\alpha$, $\beta_{xx'}=0$; but more general Hamiltonians of the form (4.1) will be generated on coarse grids, so we include them from the start.[74]

The obvious choice of coarse-grid-correction move is to rotate simultaneously all the spins in a 2^d block. The update is therefore $\theta\to\theta+p_{l,l-1}\psi$, where ψ is a coarse-grid field of angles, $p_{l,l-1}$ is the piecewise-constant injection (2.23), and addition is interpreted modulo 2π. The coarse-grid Hamiltonian

$$H_{l-1}(\psi)\equiv H_l(\phi+p_{l,l-1}\psi) \tag{4.2}$$

can easily be computed: it is

$$H_{l-1}(\psi)=\sum_{|y-y'|=1}[\alpha'_{yy'}\cos(\psi_y-\psi_{y'})+\beta'_{yy'}\sin(\psi_y-\psi_{y'})]\ , \tag{4.3}$$

where

$$\alpha'_{yy'}=\sum_{\substack{x\in B_y\\ x'\in B_{y'}}}[\alpha_{xx'}\cos(\theta_x-\theta_{x'})+\beta_{xx'}\sin(\theta_x-\theta_{x'})]\ , \tag{4.4a}$$

$$\beta'_{yy'}=\sum_{\substack{x\in B_y\\ x'\in B_{y'}}}[\beta_{xx'}\cos(\theta_x-\theta_{x'})-\alpha_{xx'}\sin(\theta_x-\theta_{x'})]\ . \tag{4.4b}$$

Here B_y is the block consisting of those 2^d sites of grid Ω_l which are affected by interpolation from the coarse-grid site $y\in\Omega_{l-1}$ [see Fig. 1(b)]. Note that the coarse-grid Hamiltonian H_{l-1} has the same functional form as the "fine-grid" Hamiltonian H_l; it is specified by the coefficients $\{\alpha'_{yy'}\}$ and $\{\beta'_{yy'}\}$. Therefore, with this definition of the interpolation operator $p_{l,l-1}$, the nonlinear multigrid algorithm (3.1) can be applied without change.[75]

This basic construction can be extended to nonlinear σ models in which the state space at each site is a *group* G. For simplicity we assume that G is a real or complex matrix group, i.e., a subgroup of $GL(n,\mathbf{R})$ or $GL(n,\mathbf{C})$ and that the Hamiltonian is of the form

$$H_l(g)=\sum_{|x-x'|=1}\mathrm{Re\,tr}(\alpha_{xx'}g_x^\dagger g_{x'})\ , \tag{4.5}$$

where $\alpha_{xx'}$ is an $n\times n$ complex matrix (not necessarily an element of G). If G is a unitary group, the case $\alpha_{xx'}=\alpha I$ corresponds to the principal chiral models,[76] while more general Hamiltonians of the form (4.5) arise in the study of Gribov copies and Landau gauge fixing.[73] [See also (4.10)–(4.13) below.] General Hamiltonians of the form (4.5) will in any case be generated on coarse grids.

The obvious choice of course-grid-correction move is to simultaneously left multiply all the spins in a $2d$ block B_y by some group element h_y. [Because of the way in which we have written Hamiltonian (4.5), left multiplication is more convenient than right multiplication.] The update is therefore $g\to(p_{l,l-1}h)g$ where h is a coarse-grid field with values in G, and $p_{l,l-1}$ is the piecewise-constant injection

$$(p_{l,l-1}h)_x=h_y\quad\text{for } x\in B_y\ . \tag{4.6}$$

The coarse-grid Hamiltonian

$$H_{l-1}(h)\equiv H_l((p_{l,l-1}h)g) \tag{4.7}$$

is therefore

$$H_{l-1}(h)=\sum_{|y-y'|=1}\mathrm{Re\,tr}(\alpha'_{yy'}h_y^\dagger h_{y'})\ , \tag{4.8}$$

where

$$\alpha'_{yy'}=\sum_{\substack{x\in B_y\\ x'\in B_{y'}}}g_{x'}\alpha_{xx'}g_x^\dagger\ . \tag{4.9}$$

The nonlinear multigrid algorithm (3.1) can then be applied with only trivial modifications (e.g., initializing the coarse-grid field to the identity matrix rather than to zero).

Finally, consider those nonlinear σ models in which the state space at each site is a *manifold* $M\subset\mathbf{R}^n$ [or $\mathbf{C}^n$] on which some group $G\subset\mathrm{GL}(n,\mathbf{R})$ [or $\mathrm{GL}(n,\mathbf{C})$] acts transitively. One example is the n-vector model, in which M is the unit sphere in $\mathbf{R}^n$ and $G=\mathrm{SO}(n)$ or $\mathrm{O}(n)$. Assume that the Hamiltonian is of the form

$$H(\phi)=\sum_{|x-x'|=1}\mathrm{Re}(a_{xx'}\phi_x^\dagger\phi_{x'})\ , \tag{4.10}$$

where $a_{xx'}$ is a real or complex number. Now fix a reference configuration $\bar\phi=\{\bar\phi_x\}_{x\in\Omega}\in M^\Omega$. The G^Ω can be mapped into M^Ω by

$$g\to g\bar\phi \tag{4.11}$$

(note that in general this map is many to one), and hence the Hamiltonian (4.10) can be "lifted" to G^Ω:

$$\tilde H(g)\equiv H(g\bar\phi)=\sum_{|x-x'|=1}\mathrm{Re\,tr}(\alpha_{xx'}g_x^\dagger g_{x'})\ , \tag{4.12}$$

where

$$\alpha_{xx'}=a_{xx'}\bar\phi_{x'}\bar\phi_x^\dagger \tag{4.13}$$

is an $n\times n$ matrix of rank 1. This Hamiltonian is of the form (4.5), and so can be handled by the multigrid method described previously.[77] Absolute minima of $\tilde H$ are in many-to-one correspondence with absolute minima of H under the map (4.11).

We mention, finally, that *combinations* of the foregoing methods can be appropriate for some models. Consider, for example, an n-component ϕ^4 model ($n\ge 2$). Then the angular variables $\omega_x\equiv\phi_x/|\phi_x|$ could be updated by a multiplicative MG algorithm, as in the corresponding n-vector model, while the magnitudes $|\phi_x|$ could be updated by nonlinear Gauss-Seidel or by additive MG. The point is that the algorithm should be adapted to the *physics* of each problem, i.e., to the relevant large-scale collective modes.

V. DETERMINISTIC MULTIGRID METHOD: LATTICE GAUGE THEORIES

In this section we give two alternative multigrid methods, developed in collaboration with Dan Zwanziger, for minimizing a Hamiltonian of the form arising in lattice gauge theories. The basic multigrid algorithm is still (3.1); the problem is to define the interpolation operators in a way that reflects the geometric (parallel-transport) properties of a gauge theory. Unfortunately, these algorithms seem to be practical only in the Abelian case; for non-Abelian theories, the coarse-grid Hamiltonian becomes very complicated. We present these methods to illustrate the general principles (which we believe are sound) and to show the difficulties which arise in the

non-Abelian case, in the hope that someone will find a way of overcoming them. This section can be skipped on a first reading.

Consider a pure lattice gauge theory with gauge group G (assumed for simplicity to be a real or complex group of unitary matrices) and Hamiltonian of the form

$$H_l(U) = \sum_{\langle xyzs \rangle} \mathrm{Re}\,\mathrm{tr}(\alpha_{xyzw} U_{xy} U_{yz} U_{zw} U_{wx}) , \qquad (5.1)$$

where α_{xyzw} is an $n \times n$ complex matrix (not necessarily an element of G) and the sum runs over all oriented plaquettes $\langle xyzw \rangle$. The case of primary physical interest is $\alpha_{xyzw} = \alpha I$, but more general Hamiltonians of the form (5.1) will in some cases be generated anyway on coarse grids.

A natural choice of course-grid-correction move (e.g., by analogy with the nonlinear σ models) is to simultaneously left multiply several parallel link variables by the "same" group element. The trouble is that, in a gauge theory, the "sameness" of group elements at different sites has no gauge-invariant meaning. Rather, one must *parallel transport* the group element from one site to another, along some predetermined path. (Note that in the presence of curvature, the parallel transport depends on the path chosen.) The idea is, therefore, to make coarse-grid-correction moves which left multiply the link variables (or a subset of them) by a field which is approximately piecewise *covariant* constant. (Note, however, that no field can be *exactly* covariant constant if the background gauge field has nonzero curvature.[78])

The simplest implementation of this idea is shown in Fig. 2 (for simplicity in the case $d = 2$). Divide the sites of the lattice, as usual, into blocks of size 2^d. Then any two neighboring blocks are connected by 2^{d-1} parallel links (illustrated by wavy lines in Fig. 2); it is these links that will be updated together. In this scheme, therefore, only half of the links in the lattice are being updated in a single coarse-grid-correction step; this is reminiscent of the trivial prolongation (2.22).

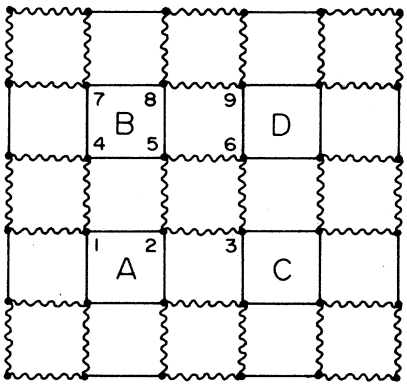

FIG. 2. Simplest implementation of multigrid for lattice gauge theories, shown here in dimension $d = 2$. The lattice is divided into blocks of size 2^d (sets of sites connected by straight lines); here A, B, C, and D are blocks. Neighboring blocks are connected by 2^{d-1} parallel links (wavy lines); these kinks are updated together.

Now let us look more closely at the meaning of a "piecewise-covariant-constant" update. Consider, for example, the links 14 and 25 in Fig. 2. Under a gauge transformation, the link variable U_{14} transforms as $U_{14} \to g_1 U_{14} g_4^{-1}$. Now imagine that we have made an update $U'_{14} = V U_{14}$. Then V transforms under the gauge group as

$$V \to g_1 V g_1^{-1} , \qquad (5.2)$$

i.e., V transforms under the *adjoint representation* of the gauge group at site 1. (In particular, if G is Abelian, then V is a gauge-invariant quantity. In other words, while the *values* of an Abelian gauge field are gauge dependent, the *changes* of those values are gauge independent.[79]) Therefore, to update link 25 we should transport the group element V to site 2—yielding $\tilde{V} \equiv U_{12}^{-1} V U_{12}$, which indeed transforms under the gauge group as $\tilde{V} \to g_2 \tilde{V} g_2^{-1}$—and then use the result to update U_{25}, i.e., $U'_{25} = \tilde{V} U_{25} \equiv U_{12}^{-1} V U_{12} U_{25}$. We emphasize that the choice of path from site 1 to site 2 is arbitrary: for a nearest-neighbor pair of sites, it is clearly simplest to choose the direct path, as we have done, but it is not mandatory to do so; any other path from site 1 to site 2 would have been just as good, e.g., we could have defined $\tilde{V} \equiv U_{25} U_{54} U_{41} V U_{41}^{-1} U_{54}^{-1} U_{25}^{-1}$. This remark is important in dimension $d \geq 3$, where there is no unique "simplest" choice of transport paths.

In summary, therefore, the scheme for updating the links joining blocks B_y and $B_{y'}$ is the following.

(1) Choose arbitrarily one of the blocks (say, B_y) and one of the points in that block which is adjacent to the other block (say, x).

(2) Assign a variable V which transforms under the adjoint representation at site x, i.e., $V \to g_x V g_x^{-1}$.

(3) Choose arbitrarily a path from x to each other point in B_y which is adjacent to $B_{y'}$, and transport V to that site using the *old* values of the gauge field $\{U\}$.

(4) Use these transported V's to left multiply the corresponding gauge fields.

One now does this for *each* pair of adjacent blocks, and considers the field of variables $\{V\}$ to be a coarse-grid field. The coarse-grid Hamiltonian is then defined to be

$$H_{l-1}(V) \equiv H_l(VU) , \qquad (5.3)$$

where we have written schematically VU to indicate the action of V on U as described above. The form of the course-grid Hamiltonian depends on the dimension of the lattice ($d = 2$ or $d \geq 3$) and on the nature of the gauge group (Abelian or non-Abelian):

There are three classes of plaquettes: those with no wavy links (e.g., 4587 in Fig. 2), those with two wavy links (e.g., 1254), and those with four wavy links (e.g., 2365). The first class obviously contributes nothing to the coarse-grid Hamiltonian. The second class contributes terms of the form

$$\begin{aligned} &\mathrm{Re}\,\mathrm{tr}[\alpha_{1254} U_{12} (U_{12}^{-1} V U_{12} U_{25}) U_{54} (V U_{14})^{-1}] \\ &\quad = \mathrm{Re}\,\mathrm{tr}[\alpha_{1254} V (U_{12} U_{25} U_{54} U_{41}) V^{-1}] \\ &\quad = \mathrm{Re}\,\mathrm{tr}[(V^{-1} \alpha_{1254} V) U_{12} U_{25} U_{54} U_{41}] . \end{aligned} \qquad (5.4)$$

If either G is Abelian or α_{1254} is a multiple of the identity matrix, then this term is a constant (independent of V). Otherwise it is a new single-link interaction of the form

$$\sum_{\langle xy \rangle} \mathrm{Re\,tr}(\beta_{xy} V_{xy} \gamma_{xy} V_{xy}^{-1}) \ , \tag{5.5}$$

where β_{xy} and γ_{xy} are $n \times n$ matrices. Such a term is perhaps tolerable, but unfortunately it generates a *sum* of such terms on coarser grids. [Moreover, in dimension $d \geq 3$ each coarse-grid link V participates $(d-1)2^{d-2}$ plaquettes of this class, so a sum of terms (5.5) is generated already on the first coarse grid.] So it seems impossible to stop short of writing the interaction as a general four-index tensor in internal space,

$$\sum_{\langle xy \rangle} \sum_{a,b,c,d=1}^{n} \mathrm{Re}[\beta_{xy}^{abcd} V_{xy}^{ab} (V_{xy}^{-1})^{cd}] \ ; \tag{5.6}$$

this requires n^4 parameters per coarse-grid link.

Finally, the third class of plaquettes (those with four wavy lines) contributes terms of the form

$$\begin{aligned} &\mathrm{Re\,tr}[\alpha_{2365}(V_{AC}U_{23})(V_{CD}U_{36})(V_{BD}U_{56})(V_{AB}U_{25})] \\ &\quad = \mathrm{Re\,tr}[\alpha_{2365} V_{AC} U_{23} V_{CD} (U_{36} U_{65}) V_{DB} U_{52} V_{BA}] \ , \end{aligned} \tag{5.7}$$

where the V's are update matrices[80]

$$U'_{23} = V_{AC} U_{23} \ , \tag{5.8a}$$

$$U'_{36} = V_{CD} U_{36} \ , \tag{5.8b}$$

$$U'_{56} = V_{BD} U_{56} \ , \tag{5.8c}$$

$$U'_{25} = V_{AB} U_{25} \ . \tag{5.8d}$$

If G is Abelian, this is again a coupling of the form (5.1), with

$$\alpha'_{ACDB} = \alpha_{2365} U_{23} U_{36} U_{65} U_{52} \ . \tag{5.9}$$

However, if G is non-Abelian, it is a mess requiring $4n^2$ parameters per coarse-grid plaquette:

$$\sum_{\langle xyzw \rangle} \mathrm{Re\,tr}(\alpha_{xyzw} U_{xy} \beta_{xyzw} U_{yz} \gamma_{xyzw} U_{zw} \delta_{xyzw} U_{wx}) \ . \tag{5.10}$$

Moreover, in dimension $d \geq 3$ each such coarse-grid plaquette participates in 2^{d-2} fine-grid plaquette, and the terms proliferate further on coarser grids, so it seems impossible to stop short of writing a general eight-index tensor in internal space, at a cost of n^8 parameters per coarse-grid plaquette.

In summary, if G is Abelian, then the coarse-grid Hamiltonian is again of the form (5.1), and the algorithm is simple and practical; its performance is currently under study.[7] If G is non-Abelian, then the simplest coarse-grid Hamiltonian that includes (5.1) and is stable under coarsening has an enormous number of parameters, making the algorithm wildly impractical, although there is nothing wrong with it in principle.

One alternative approach in the non-Abelian case would be to use a *unigrid* version of the foregoing algorithm; we explain this at the end of this section.

A second type of interpolation for lattice gauge theories is shown in Fig. 3 (again in $d=2$ for simplicity). Here a "coarse-grid link" is associated with 2^{d-1} parallel *pairs* of links (each pair consists of two links connected "in series"); four such coarse-grid links (V, W, X, and Y) are depicted in Fig. 3. In this scheme, therefore, *all* of the links in the lattice are updated in a single coarse-grid correction step; this mimics the piecewise-constant interpolation (2.23).

Let us assign the following transformation laws under the gauge group:

$$V \to g_4 V g_4^{-1} \ , \tag{5.11a}$$

$$W \to g_2 W g_2^{-1} \ , \tag{5.11b}$$

$$X \to g_6 X g_6^{-1} \ , \tag{5.11c}$$

$$Y \to g_8 Y g_8^{-1} \ . \tag{5.11d}$$

Then the fine-grid links are updated as follows:

$$U'_{14} = (U_{14} V U_{14}^{-1}) U_{14} = U_{14} V \ , \tag{5.12a}$$

$$U'_{47} = V U_{47} \ , \tag{5.12b}$$

$$U'_{25} = U_{25} U_{45}^{-1} V U_{45} \ , \tag{5.12c}$$

$$U'_{58} = U_{45}^{-1} V U_{45} U_{58} \ , \tag{5.12d}$$

$$U'_{12} = (U_{12} W U_{12}^{-1}) U_{12} = U_{12} W \ , \tag{5.12e}$$

$$U'_{23} = W U_{23} \ , \tag{5.12f}$$

$$U'_{36} = (U_{36} X U_{36}^{-1}) U_{36} = U_{36} X \ , \tag{5.12g}$$

$$U'_{69} = X U_{69} \ , \tag{5.12h}$$

$$U'_{45} = U_{45} U_{58} Y U_{58}^{-1} \ , \tag{5.12i}$$

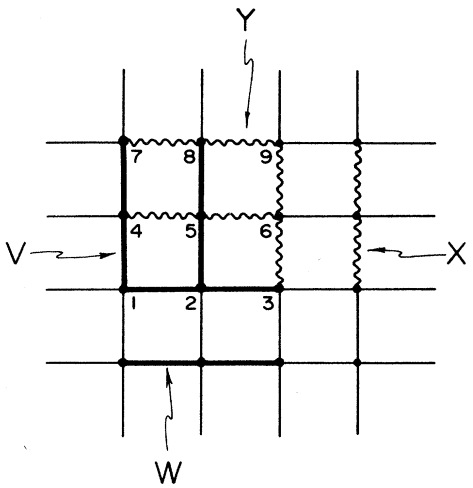

FIG. 3. Alternative implementation of multgrid for lattice gauge theories, shown here in dimension $d=2$. A "coarse-grid link" is associated with 2^{d-1} parallel pairs of links (each pair consists of two links connected "in series"). Four coarse-grid links are depicted here: V (vertical solid lines), W (horizontal solid lines), X (vertical wavy lines), and Y (horizontal wavy lines).

$$U'_{56} = U_{58} Y U_{58}^{-1} U_{56} , \tag{5.12j}$$

$$U'_{78} = (U_{78} Y U_{78}^{-1}) U_{78} = U_{78} Y , \tag{5.12k}$$

$$U'_{89} = Y U_{89} . \tag{5.12l}$$

(Of course, many other choices are possible.) Then there are three classes of plaquettes: those involving two distinct coarse-grid links (e.g., 4587), those involving three distinct coarse-grid links (e.g., 1254 or 5698), and those involving four distinct coarse-grid links (e.g., 2365). The first class contributes terms of the form

$$\mathrm{Re\,tr}[\alpha_{4587}(U_{45} U_{58} Y U_{58}^{-1})(U_{45}^{-1} V U_{45} U_{58}) \times (Y^{-1} U_{78}^{-1})(U_{47}^{-1} V^{-1})] . \tag{5.13}$$

If G is Abelian, then this term is a constant (independent of V and Y); otherwise it is a mess. The second class of plaquettes contributes terms of the form

$$\mathrm{Re\,tr}[\alpha_{1254}(U_{12} W)(U_{25} U_{45}^{-1} V U_{45}) \times (U_{58} Y^{-1} U_{58}^{-1} U_{45}^{-1})(V^{-1} U_{14}^{-1})] . \tag{5.14}$$

If G is Abelian, then this is of the form $\mathrm{Re\,tr}(MWY^{-1})$ with $M = \alpha_{1254} U_{12} U_{25} U_{45}^{-1} U_{14}^{-1}$; that is, it is a new interaction between parallel links on opposite sides of a plaquette,

$$\sum_{\langle xyzw \rangle} \mathrm{Re\,tr}(\beta_{xy;zw} V_{xy} V_{zw}) , \tag{5.15}$$

where $\beta_{xy;zw}$ is an $n \times n$ matrix. Note that such a term produces only terms of the same form on coarser grids. On the other hand, if G is non-Abelian, then (5.14) is a mess. Finally, the third class of plaquettes contributes terms of the form

$$\mathrm{Re\,tr}[\alpha_{2365}(W U_{23})(U_{36} X)(U_{56}^{-1} U_{58} Y^{-1} U_{58}^{-1}) \times (U_{45}^{-1} V^{-1} U_{45} U_{25}^{-1})] . \tag{5.16}$$

If G is Abelian, this is again a coupling of the form (5.1), with

$$\alpha'_{WXY^{-1}Z^{-1}} = \alpha_{2365} U_{23} U_{36} U_{56}^{-1} U_{25}^{-1} \tag{5.17}$$

If G is non-Abelian, it is a mess.

One way of simplifying these formulas in the non-Abelian case (for either of the two interpolations) is to update only *one* direction of links at a time. Suppose, for example, that we update only the vertical links. Then, in each fine-grid plaquette, at most *two* links are being updated: these involve either the same coarse-grid link twice (e.g., 1254), or two neighboring coarse-grid links (e.g., 2365). Such plaquettes produce coarse-grid interactions of the form (5.6) and (5.15), respectively; and this form of the Hamiltonian is preserved on coarser grids. This coarse-grid Hamiltonian requires, therefore, n^4 parameters per coarse-grid link and plaquette. Though obviously unwieldy, it might be borderline practical.

A closely related approach would be to use a *unigrid* updating. Here one would update simultaneously, for example, all the vertical links in a box B, using a generalization of the "piecewise-covariant-constant" interpolation: the update matrix V would be parallel transported along a maximal tree in B. The resulting "unigrid Hamiltonian" is obviously of the form

$$H_{\mathrm{UG}}(V) = \mathrm{Re\,tr}(\alpha V) + \sum_{a,b,c,d=1}^{n} \mathrm{Re}[\beta^{abcd} V^{ab} (V^{-1})^{cd}] \tag{5.18}$$

for some matrix α and fourth-rank tensor β. Moreover, the coefficients α and β can be computed in a CPU time proportional to the volume of the box, by carrying out the parallel transport successively along the tree.[81] Again, this might be borderline practical.

Finally, we remark that lattice gauge theories with bosonic matter fields can be handled by a generalization of the above formulas; we leave the details as an exercise for the reader. The resulting algorithms appear, however, to be practical only in the Abelian case.

VI. DYNAMIC MONTE CARLO METHODS: A REVIEW

In this section we review briefly the principles of dynamic Monte Carlo methods, and define some quantities (autocorrelation times) which will play an important role in the remainder of the paper.

Monte Carlo methods can be classified as *static* or *dynamic*. Static methods are those that generate a sequence of *statistically independent* samples from the desired probability distribution π. Dynamic methods are those that generate a sequence of *correlated* samples from some stochastic process (usually a Markov process) having the desired probability distribution π as its unique equilibrium distribution.

More precisely, let S be the configuration space of the system; we use the letter ϕ to denote a generic configuration. Now consider a Markov chain with state space S and transition probability kernel[82] $P(\phi \to \phi')$ satisfying the following two conditions.

(a) $\int d\pi(\phi) P(\phi \to \phi') = d\pi(\phi')$. (6.1)

[This condition says that P leaves invariant the probability distribution π; in other words, π is an *invariant* (or *stationary* or *equilibrium)* distribution for P.]

(b) For each $\phi \in S$ and each $A \subset S$ having $\pi(A) > 0$, there exists an $n \geq 0$ for which $P^n(\phi, A) > 0$. Here P^n is the n-step transition probability, i.e., the nth power of the kernel P. (This condition is called *irreducibility*;[83] it asserts, roughly speaking, that each state can eventually be reached from each other state.)

In this case it can be shown[84] that π is the *unique* stationary distribution for the Markov chain with transition probability P, and that the occupation-time distribution over long time intervals converges (with probability 1) to π, irrespective of the initial state of the system. If, in addition, P is *aperiodic* [this means that for each pair ϕ, A, we have $P^n(\phi, A) > 0$ for *all* sufficiently large n], then the probability distribution at any single time in the far future also converges to π, irrespective of the initial state—that is, $\lim_{n \to \infty} P^n(\phi, \cdot) = \pi$ for all ϕ. Thus, simulation of the Markov chain P provides a legitimate Monte

Carlo method for estimating averages with respect to π.

However, since the successive states $\phi^{(0)},\phi^{(1)},\ldots$ of this Markov chain are in general highly correlated, the variance of estimates produced in this way may be much higher than in independent sampling. To make this precise, let F be a real-valued function defined on the state space S (i.e., a real-valued observable) which is square integrable with respect to π; and consider the *stationary* Markov chain (i.e., start the system in the stationary distribution π, or equivalently, "thermalize" it for a very long time prior to observing the system). Then $\{F_t\}\equiv\{F(\phi^{(t)})\}$ is a stationary stochastic process with mean

$$\mu_F\equiv\langle F_t\rangle=\int d\pi(\phi)F(\phi) \tag{6.2}$$

and *unnormalized autocorrelation function*[85]

$$\begin{aligned}C_{FF}(t)&\equiv\langle(F_s-\langle F_s\rangle)(F_{s+t}-\langle F_{s+t}\rangle)\rangle\\&=\langle F_sF_{s+t}\rangle-\mu_F^2\\&=\int d\pi(\phi)[P^{|t|}(\phi\to\phi')-d\pi(\phi')]F(\phi)F(\phi')\ .\end{aligned} \tag{6.3}$$

The *normalized autocorrelation function* is then

$$\rho_{FF}(t)\equiv C_{FF}(t)/C_{FF}(0)\ . \tag{6.4}$$

Now define the *integrated autocorrelation time* (for the observable F) to be

$$\begin{aligned}\tau_{\mathrm{int},F}&\equiv\tfrac{1}{2}\sum_{t=-\infty}^{\infty}\rho_{FF}(t)\\&=\tfrac{1}{2}+\sum_{t=1}^{\infty}\rho_{FF}(t)\end{aligned} \tag{6.5}$$

[The factor of $\frac{1}{2}$ is purely a matter of convention; it is inserted so that $\tau_{\mathrm{int},F}\approx\tau_{\mathrm{exp},F}$ if $\rho_{FF}(t)\approx e^{-|t|/\tau}$ with $\tau\gg1$.] The relevance of the integrated autocorrelation time is that it controls the statistical error in Monte Carlo measurements of $\langle F\rangle$. More precisely, the sample mean

$$\bar F\equiv\frac{1}{n}\sum_{t=1}^{n}F_t \tag{6.6}$$

has variance

$$\begin{aligned}\mathrm{var}(\bar F)&=\frac{1}{n^2}\sum_{r,s=1}^{n}C_{FF}(r-s)\\&=\frac{1}{n}\sum_{t=-(n-1)}^{n-1}(1-|t|/n)C_{FF}(t)\end{aligned} \tag{6.7a}$$

$$\approx\frac{1}{n}(2\tau_{\mathrm{int},F})C_{FF}(0)\quad\text{for } n\gg\tau\ . \tag{6.7b}$$

Thus, the variance of $\bar F$ is a factor $2\tau_{\mathrm{int},F}$ larger than it would be if the $\{F_t\}$ were statistically independent. Stated differently, the number of "effectively independent samples" in a run of length n is roughly $n/2\tau_{\mathrm{int},F}$.

A second (and somewhat different) notion of autocorrelation time plays some role in theoretical analyses. Since typically $\rho_{FF}(t)$ decays exponentially ($\sim e^{-|t|/\tau}$) for large t, we define the *exponential autocorrelation time*

$$\tau_{\mathrm{exp},F}=\limsup_{t\to\infty}\frac{t}{-\ln|\rho_{FF}(t)|} \tag{6.8}$$

and

$$\tau_{\mathrm{exp}}=\sup_{F\in L^2(\pi)}\tau_{\mathrm{exp},F}\ . \tag{6.9}$$

Thus, τ_{exp} is the relaxation time of the slowest mode in the system. (If the state space is infinite, τ_{exp} might be $+\infty$.) An equivalent definition, which is useful for rigorous analysis, involves considering the transition probability kernel P as an operator on the Hilbert space $L^2(\pi)$ of square-integrable observables, defined by

$$(Pf)(\phi)=\int P(\phi\to d\phi')f(\phi')\ . \tag{6.10}$$

It is not hard to show that P is a contraction (i.e., has norm ≤1), so that its spectrum lies in the unit disk. The constant function 1 is a nondegenerate eigenvector of P (and of its adjoint P^*) with eigenvalue 1; and if P is aperiodic, then this is the only eigenvalue on the unit circle.[86] Let R be the spectral radius of the remainder of P, i.e.,

$$R\equiv\inf\{r:\mathrm{spec}P\subset\{\lambda:|\lambda|\le r\}\cup\{1\}\}\ . \tag{6.11}$$

Then it is not difficult to show, using a generalization of the spectral radius formula,[87] that $R=\exp(-1/\tau_{\mathrm{exp}})$. In particular, the rate of convergence to equilibrium from an initial nonequilibrium distribution is controlled by R, and hence by τ_{exp}.

In summary, the autocorrelation times τ_{exp} and $\tau_{\mathrm{int},F}$ play different roles in Monte Carlo simulations. τ_{exp} provides an upper bound (sometimes much too conservative) on the number of iterations n_{disc} which should be discarded at the beginning of the run, before the system has attained equilibrium; for example, $n_{\mathrm{disc}}\gtrsim20\tau_{\mathrm{exp}}$ is usually more than adequate. On the other hand, $\tau_{\mathrm{int},F}$ determines the statistical errors in Monte Carlo measurements of $\langle F\rangle$, once equilibrium has been attained.

Finally, we note that one convenient way of satisfying condition (a) is to satisfy the following *stronger* condition:

$$\text{(a')}\quad d\pi(\phi)P(\phi\to\phi')=d\pi(\phi')P(\phi'\to\phi)\ . \tag{6.12}$$

Summing (a') over ϕ, we recover (a). (a') is called the *detailed-balance* condition;[88] it is equivalent to the *self-adjointness* of P as an operator on the space $L^2(\pi)$. In this case, it follows from the spectral theorem that the autocorrelation function $C_{FF}(t)$ has a spectral representation

$$C_{FF}(t)=\int_{-1}^{1}d\rho(\lambda)\lambda^{|t|}\ , \tag{6.13}$$

where $d\rho(\lambda)$ is a *positive* measure. Moreover, we have

$$R=e^{-1/\tau_{\mathrm{exp}}}=\sup\{|\lambda|:\lambda\in\mathrm{sup}\rho\} \tag{6.14}$$

and

$$\tau_{\mathrm{int},F}=\tfrac{1}{2}\int_{-1}^{1}d\rho(\lambda)\frac{1+\lambda}{1-\lambda} \tag{6.15a}$$

$$\le\frac{1}{2}\frac{1+e^{-1/\tau_{\mathrm{exp}}}}{1-e^{-1/\tau_{\mathrm{exp}}}}\approx\tau_{\mathrm{exp}}\ . \tag{6.15b}$$

We want to emphasize the occasionally overlooked fact that *detailed balance is not necessary for the correctness of a Monte Carlo algorithm* —all that is necessary is to satisfy conditions (a) and (b). Indeed, many widely used Monte Carlo algorithms do *not* satisfy detailed balance. For example, if P_1 and P_2 are two transition matrices that preserve the distribution π [i.e., satisfy condition (a)], then their composition P_1P_2 clearly also preserves π. On the other hand, the composition of two transition matrices, each of which satisfies detailed balance, does *not* in general satisfy detailed balance: in fact, the product of two self-adjoint operators is self-adjoint *if and only if the two operators commute.* Thus, the standard heat-bath algorithm, with a periodic sweep of the sites, does *not* satisfy detailed balance. But it leaves invariant the Gibbs measure π, which is all that counts.

VII. MULTIGRID MONTE CARLO ALGORITHM

Classical equilibrium statistical mechanics is a natural generalization of classical statics (for problems posed in variational form): in the latter we seek to minimize a Hamiltonian $H(\phi)$, while in the former we seek to generate random samples from the Boltzmann-Gibbs probability distribution $e^{-\beta H(\phi)}$. The statistical-mechanical problem reduces to the deterministic one in the zero-temperature limit $\beta \to +\infty$.

Likewise, many (but not all) of the deterministic iterative algorithms for minimizing $H(\phi)$ can be generalized to stochastic iterative algorithms—that is, dynamic Monte Carlo methods—for generating random samples from $e^{-\beta H(\phi)}$. For example, the stochastic generalization of the Gauss-Seidel algorithm (or more generally, nonlinear Gauss-Seidel algorithm with exact minimization) is the single-site heat-bath algorithm; and the stochastic generalization of the multigrid algorithm is the multigrid Monte Carlo algorithm.

Let us explain these correspondences in more detail. In the Gauss-Seidel algorithm, the grid points are swept in some order, and at each stage the Hamiltonian is minimized as a function of a single variable ϕ_x, with all other variables $\{\phi_y\}_{y\neq x}$ being held fixed. The single-site heat-bath algorithm has the same general structure, but the new value ϕ'_x is chosen randomly from the conditional distribution of $e^{-\beta H(\phi)}$ given $\{\phi_y\}_{y\neq x}$, i.e., from the one-dimensional probability distribution

$$P(\phi'_x)d\phi'_x = \text{const} \times \exp[-\beta H(\phi'_x, \{\phi_y\}_{y\neq x})]d\phi'_x \qquad (7.1)$$

(where the normalizing constant depends on $\{\phi_y\}_{y\neq x}$). It is not difficult to see that this operation leaves invariant the Gibbs distribution $e^{-\beta H(\phi)}$. As $\beta \to +\infty$ it reduces to the Gauss-Seidel algorithm.

It is useful to visualize geometrically the action of the Gauss-Seidel and heat-bath algorithms within the space U of all possible field configurations. Starting at the current field configuration ϕ, the Gauss-Seidel and heat-bath algorithms propose to move the system along the line in U consisting of configurations of the form $\phi' = \phi + t\delta_x (-\infty < t < \infty)$, where δ_x denotes the configuration which is 1 at site x and 0 elsewhere. In the Gauss-Seidel algorithm, t is chosen so as to minimize the Hamiltonian restricted to the given line; while in the heat-bath algorithm, t is chosen randomly from the conditional distribution of $e^{-\beta H(\phi)}$ restricted to the given line: namely, the one-dimensional distribution with probability density $P_{\text{cond}}(t) \sim \exp[-H_{\text{cond}}(t)] \equiv \exp[-H(\phi + t\delta_x)]$.

We now propose to generalize this idea in two ways.

(1) The "fibers" used by the algorithm need not be lines, but can be higher-dimensional linear or even nonlinear manifolds.

(2) The new configuration ϕ' need not be chosen *independently* of the old configuration ϕ (as in the heat-bath algorithm); rather, it can be selected by any updating procedure which leaves invariant the conditional probability distribution of $e^{-\beta H(\phi)}$ restricted to the fiber.

Before embarking on heavy formalism, let us give four examples to illustrate the basic idea.

Examples. (1) *Linear-state-space unigrid method* (Secs. II C and III). Here the fibers are lines of the form $\phi' = \phi + t\chi_B$ $(-\infty < t < \infty)$, where χ_B denotes the function which is 1 for sites belonging to the block B and zero elsewhere. The sets B are taken successively to be single sites, cubes of side 2, cubes of side 4, and so on. (If linear interpolation were used, then the "direction vectors" χ_B would be replaced by triangular waves of various widths.) The deterministic unigrid algorithm chooses t so as to minimize the "conditional Hamiltonian" $H_{\text{cond}}(t) \equiv H(\phi + t\chi_B)$, while the stochastic unigrid algorithm chooses t randomly from the one-dimensional distribution with probability density $P_{\text{cond}}(t) \sim \exp[-H_{\text{cond}}(t)]$. Conceptually this algorithm is no more complicated than the single-site heat-bath algorithm. But physically it is of course very different, as the direction vectors χ_B represent *collective modes* on all length scales.

(2) *Nonlinear-σ-model unigrid method* (Secs. II C and IV). Consider a nonlinear σ model with values in a compact group G. Here the variable t is an element of G, and the action of t on the spin configuration g is to left multiply all the spins in a block B by the group element t:

$$g'_x = t \circ_B g \equiv \begin{cases} tg_x & \text{if } x \in B \ , \\ g_x & \text{if } x \notin B \ . \end{cases} \qquad (7.2)$$

Again the sets B are taken successively to be single sites, cubes of side 2, cubes of side 4, and so on. The deterministic unigrid algorithm chooses t so as to minimize the conditional Hamiltonian $H_{\text{cond}}(t) \equiv H(t \circ_B g)$, while the stochastic unigrid algorithm chooses t randomly from the distribution on G whose density with respect to Haar measure is $P_{\text{cond}}(t) \sim \exp[-H_{\text{cond}}(t)]$. Because of the curvature of the fibers in this example, it is not entirely obvious that this algorithm leaves invariant the correct Gibbs distribution (one might worry that there are missing Jacobian factors)—but it is true, as we prove below.

(3) *Linear-state-space multigrid method* (Secs. II and III). Here the fibers are the sets of field configurations which can be obtained one from another by a coarse-grid-correction step, i.e., the sets of fields $\phi + p_{l,l-1}\psi$ with ϕ fixed and ψ varying over U_{l-1}. These fibers form a family of parallel affine subspaces in U_l, of dimension $N_{l-1} = \dim U_{l-1}$. This example differs from the unigrid

examples in that the new configuration ϕ' is *not* chosen independently of the old configuration ϕ (to do so would be impractical on a space of such high dimension). Rather, ϕ' is chosen by a *valid updating procedure.* In the deterministic case, "validity" means that the updating procedure is an iterative algorithm which converges (at least locally) to the absolute minimizer of H_l restricted to the fiber. In fact, we take this iterative algorithm to be multigrid itself, applied at level $l-1$ to the conditional Hamiltonian $H_{l-1}(\psi) \equiv H_l(\phi + p_{l,l-1}\psi)$. Validity of the multigrid algorithm can therefore be proven inductively, starting at level 0 and working upwards. In the stochastic (MGMC) case, "validity" means that the updating procedure is a Markov chain which leaves invariant the conditional distribution of $e^{-\beta H_l(\phi)} d\phi_l$ restricted to the fiber. But this distribution, written in the coordinates ψ, is just $e^{-\beta H_{l-1}(\psi)} d\psi$. (The point is that Lebesgue measure is preserved, up to a multiplicative constant, by the affine transformation $\psi \to \phi + p_{l,l-1}\psi$.) It is therefore justified to use, as the updating procedure, the MGMC algorithm itself, applied at level $l-1$ to the conditional Hamiltonian $H_{l-1}(\psi)$. Validity of the MGMC algorithm can be proven inductively, starting at level 0 (where the MGMC algorithm is just ordinary heat bath) and working upwards.

Of course, this MGMC algorithm is *mathematically and physically equivalent* to the stochastic unigrid algorithm described in Example (1). But it is useful, we believe, to be able to look at it from either of the two points of view: independent resamplings in one-dimensional fibers, or nonindependent resamplings (defined recursively) in higher-dimensional (coarse-grid) fibers. On the other hand, the two algorithms are not *computationally* equivalent. One MGMC sweep requires a CPU time of the order of the volume (provided that $\gamma < 2^d$), while the time for a unigrid sweep grows faster than the volume [cf. the work estimates (2.30) and (2.42)].

(4) *Nonlinear-σ-model multigrid method* (Sec. IV). The fibers are again the sets of fields which can be obtained one from another by a coarse-grid-correction step, i.e., the sets of fields $(p_{l,l-1}h)g$ with $g \in U_l$ fixed and h varying over U_{l-1}. The comments made in the preceding example apply here too. The only difference arises from the curvature of the fibers, so that the proof of validity of the stochastic algorithm is not entirely trivial.

With this introduction, let us now define the general concept underlying all these examples, which we call *partial resampling.* It goes as follows: Let the configuration space U be decomposed as a disjoint union of "fibers" U_α, where α runs over some (usually continuous) index set $\mathcal{A}$. Then any probability measure μ on U can be written in the form

$$d\mu(\phi) = \int d\nu(\phi|\alpha) d\rho(\alpha) , \tag{7.3}$$

where ρ is a probability measure on $\mathcal{A}$, and $d\nu(\cdot|\alpha)$ is, for each α, a probability measure on U_α. In fact, ρ is just the projection of μ onto $\mathcal{A}$—that is, the probability distribution describing *which fiber* ϕ lies in (integrating out the locations within the fibers)—while $d\nu(\cdot|\alpha)$ is the conditional probability distribution of $d\mu(\phi)$ given that ϕ lies in the fiber U_α (Ref. 89). Now let $P = P(\phi \to \phi')$ be any transition probability kernel which leaves invariant each of the measures $d\nu(\cdot|\alpha)$, i.e., which satisfies

$$\int d\nu(\phi|\alpha) P(\phi \to \phi') = d\nu(\phi'|\alpha) \tag{7.4}$$

for all α. Then it follows from (7.3) that P also leaves μ invariant, i.e.,

$$\int d\mu(\phi) P(\phi \to \phi') = d\mu(\phi') . \tag{7.5}$$

Physically, P is an updating procedure which moves the system stochastically *within* its current fiber U_α, in such a way as to leave invariant the measure $d\nu(\cdot|\alpha)$. In summary, *any updating procedure which leaves invariant all of the conditional probability distributions $d\nu(\cdot|\alpha)$ also leaves invariant the parent probability distribution $d\mu$.*

The simplest example is that in which the configuration space U is decomposed as a Cartesian product $U = V \times W$, so that each field configuration $\phi \in U$ can be uniquely written as $\phi = (\psi, \omega)$ with $\psi \in V$, $\omega \in W$. Then we can take $\mathcal{A} = W$ and $U_\omega = V \times \{\omega\} \equiv \{(\psi, \omega) : \psi \in V\}$. Now suppose that

$$d\mu(\phi) = Z^{-1} e^{-\beta H(\psi,\omega)} d\mu_0^V(\psi) d\mu_0^W(\omega) , \tag{7.6}$$

where μ_0^V and μ_0^W are suitable *a priori* measures, and we have written $H(\psi, \omega)$ for $H(\phi)$. Then the conditional probability distribution of ψ given ω has the simple form

$$d\nu(\psi|\omega) = Z(\omega)^{-1} e^{-\beta H(\psi,\omega)} d\mu_0^V(\psi) , \tag{7.7}$$

where

$$Z(\omega) = \int e^{-\beta H(\psi,\omega)} d\mu_0^V(\psi) . \tag{7.8}$$

A special case of this setup is that in which U is a vector space decomposed as a direct sum $U = V \oplus W$. Then the fibers U_ω are a family of parallel affine subspaces. Usually μ_0^V and μ_0^W are just Lebesgue measure on V and W, respectively.

Example. In single-site update algorithms, when site x is being updated one takes $\psi = \phi_x$ and $\omega = \{\phi_y\}_{y \neq x}$.

The transition probability $P(\phi \to \phi')$ can be any updating procedure which leaves invariant each of the conditional probability distributions $d\nu(\cdot|\alpha)$. One example is *(generalized) heat-bath updating,* in which the new configuration ϕ' is an *independent* resampling of location within the fiber $\alpha = \alpha(\phi)$:

$$P_{\mathrm{HB}}(\phi \to \phi') = d\nu(\phi'|\alpha(\phi)) . \tag{7.9}$$

In the Cartesian-product situation, this means that $\phi' = (\psi', \omega)$, where ψ' is a random variable chosen from the conditional distribution (7.7), *independent* of the "old" value ψ, while ω is unchanged. Heat-bath updating is feasible if the distribution $d\nu(\cdot|\alpha)$ is relatively simple. This typically happens if the fibers U_α are low-dimensional manifolds (as in our two unigrid examples). However, we emphasize that it is legitimate to use *any* updating procedure that leaves invariant the conditional distribution $d\nu(\phi'|\alpha(\phi))$; it is not necessary that ϕ' be (conditionally) independent of ϕ. For example, Metropolis updating is perfectly legitimate—as is MGMC updating.

We can now describe the multigrid Monte Carlo (MGMC) algorithm in detail. Consider first the case in which the configuration space U is a *vector space*. We specify the following ingredients (compare to Sec. III).

(1) A sequence of *course-grid configuration spaces* $U_M \equiv U, U_{M-1}, U_{M-2}, \ldots, U_0$.

(2) *Prolongation* (or "interpolation") *operators* $p_{l,l-1}$: $U_{l-1} \to U_l$ for $1 \le l \le M$.

(3) *Basic stochastic iterations* $\mathcal{S}_l$: $U_l \times \mathcal{H}_l \to U_l$ for $0 \le l \le M$. Here $\mathcal{H}_l$ is a space of "possible Hamiltonians" defined on U_l, and $\mathcal{S}_l(\cdot, H_l)$ is a stochastic updating procedure which leaves invariant the Gibbs distribution $e^{-\beta H_l}$:

$$\int d\phi_l e^{-\beta H_l(\phi_l)} P_{\mathcal{S}_l(\cdot,H_l)}(\phi_l \to \phi_l') = d\phi_l' e^{-\beta H_l(\phi_l')} , \qquad (7.10)$$

where $d\phi_l$ is Lebesgue measure on U_l. For example, $\mathcal{S}_l(\cdot, H_l)$ could be a single-site heat-bath (or single-site Metropolis) updating for the Hamiltonian H_l. Most generally, we shall use two basic stochastic iterations, $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$; they may be the same, but need not be.

(4) *Cycle control parameters* (integers) $\gamma_l \ge 1$ for $1 \le l \le M$, which control the number of times that the coarse grids are visited.

The MGMC algorithm is then defined recursively as follows:

procedure $mgmc(l, \phi, H_l)$

comment This algorithm updates a field configuration ϕ in such a way as to leave invariant the Gibbs distribution $e^{-\beta H_l(\phi)} d\phi$, where $d\phi$, is Lebesgue measure on U_l .

$\phi \leftarrow \mathcal{S}_l^{\text{pre}}(\phi, H_l)$

if $l > 0$ **then**

compute $H_{l-1}(\cdot) \equiv H_l(\phi + p_{l,l-1}\cdot)$

$\psi \leftarrow 0$

for $j = 1$ **until** γ_l **do** $mgmc(l-1, \psi, H_{l-1})$

$\phi \leftarrow \phi + p_{l,l-1}\psi$

endif

$\phi \leftarrow \mathcal{S}_l^{\text{post}}(\phi, H_l)$

end (7.11)

It is immediately seen that this algorithm is *identical* in structure to the deterministic multigrid algorithm (3.1)—only the meaning of $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$ is different.

As explained above, the course-grid-correction phase of the MGMC algorithm is a special case of partial resampling: the configuration space U_l is decomposed as a direct sum $U_l = V_l \oplus W_l$, where $V_l \equiv p_{l,l-1}[U_{l-1}]$ and W_l is some complementary subspace (the choice of W_l plays no role). In other words, the fibers U_α are the sets of field configurations which can be obtained one from another by a coarse-grid-correction step: namely, $\phi + p_{l,l-1}\psi$ with ϕ fixed and ψ varying over U_{l-1}. These fibers form a family of parallel affine subspaces in U_l. The conditional distributions of $e^{-\beta H_l(\phi_l)} d\phi_l$ on these fibers are simply $e^{-\beta H_l(\phi_l)}$ multiplied by Lebsgue measure on the fiber; but this, in new coordinates, is precisely $e^{-\beta H_{l-1}(\psi)} d\psi$, since Lebesgue measure is preserved, up to a multiplicative constant, by the affine transformation $\psi \to \phi + p_{l,l-1}\psi$. This proves the correctness of the MGMC algorithm (7.11).

Consider next the case of a nonlinear σ model with values in a *compact group* G (cf. Sec. IV). The MGMC algorithm is again identical in structure to the corresponding deterministic MG algorithm, except for the meaning of $\mathcal{S}_l^{\text{pre}}$ and $\mathcal{S}_l^{\text{post}}$. The fibers U_α are again the sets of fields which can be obtained one from another by a coarse-grid-correction step: namely, $(p_{l,l-1}h)g$ with $g \in U_l$ fixed on h varying over U_{l-1}. It seems intuitively clear that the MGMC algorithm for nonlinear σ models is "correct," i.e., that it preserves the correct Gibbs distribution. However, it is extremely easy to make mistakes in this area (see Sec. X A), so we feel more confident with a formal proof. To simplify the notation, we will discuss a simple model situation in which there is only *one* coarse-grid site coupled to *two* fine-grid sites. The general situation with arbitrarily many coarse-grid sites (not just one) and arbitrarily many fine-grid sites per block (not just two) is proven by identical reasoning using heavier notation (more indices).

Consider, therefore, a model with fine-grid variables g_1, g_2 belonging to a compact group G. Let the Hamiltonian be $H(g_1, g_2)$; the Gibbs probability distribution is

$$d\mu(g_1, g_2) = Z^{-1} e^{-H(g_1,g_2)} d\lambda(g_1) d\lambda(g_2) , \qquad (7.12)$$

where $d\lambda$ is the normalized Haar measure[90] on G and of course Z is the normalization factor ("partition function")

$$Z = \int\int e^{-H(g_1,g_2)} d\lambda(g_1) d\lambda(g_2) . \qquad (7.13)$$

Now suppose that we put g_1 and g_2 into a "coarse-grid block" and left multiply both of them by a "coarse-grid variable" $h \in G$:

$$(g_1, g_2) \to (hg_1, hg_2) . \qquad (7.14)$$

We want to show that if (g_1, g_2) have the distribution (7.12) and h has the coarse-grid conditional distribution

$$d\mu(h | g_1, g_2) = Z(g_1, g_2)^{-1} e^{-H(hg_1,hg_2)} d\lambda(h) , \qquad (7.15)$$

where

$$Z(g_1, g_2) = \int e^{-H(hg_1,hg_2)} d\lambda(h) , \qquad (7.16)$$

then $(g_1', g_2') \equiv (hg_1, hg_2)$ has again the distribution (7.12).

To show that two probability distributions are equal, it suffices to show that they give equal expectation to all functions. So let $F(g_1, g_2)$ be an arbitrary function of g_1 and g_2 (i.e., an arbitrary "observable"). We want to show that $I(F) = J(F)$, where

$$I(F)\equiv Z^{-1}\int\int e^{-H(g_1,g_2)}F(g_1,g_2)d\lambda(g_1)d\lambda(g_2) \tag{7.17}$$

and

$$J(F)\equiv Z^{-1}\int\int e^{-H(g_1,g_2)}\left[Z(g_1,g_2)^{-1}\int e^{-H(hg_1,hg_2)}F(hg_1,hg_2)d\lambda(h)\right]d\lambda(g_1)d\lambda(g_2)\ . \tag{7.18}$$

Interchanging the order of integration in J gives

$$J(F)=Z^{-1}\int\left[\int\int Z(g_1,g_2)^{-1}e^{-H(g_1,g_2)}e^{-H(hg_1,hg_2)}F(hg_1,hg_2)d\lambda(g_1)d\lambda(g_2)\right]d\lambda(h)\ . \tag{7.19}$$

Now, Haar measure is preserved under the change of variables $g_1'=hg_1$, $g_2'=hg_2$ in the inner integral, so

$$J(F)=Z^{-1}\int\left[\int\int Z(h^{-1}g_1',h^{-1}g_2')^{-1}e^{-H(h^{-1}g_1',h^{-1}g_2')}e^{-H(g_1',g_2')}F(g_1',g_2')d\lambda(g_1')d\lambda(g_2')\right]d\lambda(h)\ . \tag{7.20}$$

Returning now to the original order of integration (and dropping the primes), we get

$$J(F)=Z^{-1}\int\int\left[\int Z(h^{-1}g_1,h^{-1}g_2)^{-1}e^{-H(h^{-1}g_1,h^{-1}g_2)}d\lambda(h)\right]e^{-H(g_1,g_2)}F(g_1,g_2)d\lambda(g_1)d\lambda(g_2)\ . \tag{7.21}$$

Now $Z(h'g_1,h'g_2)=Z(g_1,g_2)$ for any $h'\in G$, by the left invariance of Haar measure [cf. (7.16)]. Therefore,

$$J(F)=Z^{-1}\int\int\left[Z(g_1,g_2)^{-1}\int e^{-H(h^{-1}g_1,h^{-1}g_2)}d\lambda(h)\right]e^{-H(g_1,g_2)}F(g_1,g_2)d\lambda(g_1)d\lambda(g_2)\ . \tag{7.22}$$

On the other hand, Haar measure (on a *compact* group) is also invariant under inversion, i.e.,

$$\int f(h^{-1})d\lambda(h)=\int f(h)d\lambda(h) \tag{7.23}$$

for any function f, so the inner integral in (7.22) collapses to the constant 1, and $I(F)=J(F)$ as desired. This proves the correctness of the MGMC algorithm of the group-valued nonlinear σ model.

Consider, next the case of a nonlinear σ model with values in a *compact manifold* $M\subset\mathbf{R}^n$ (or $\mathbf{C}^n$) on which a compact group G acts transitively. As explained in Sec. V such a model on M can be "lifted" to a model on G by means of the mapping

$$g\to g\bar{\phi}\ , \tag{7.24}$$

where $\bar{\phi}$ is a fixed reference configuration. Of course, this mapping is usually many to one, but that is irrelevant. The key fact is that the image under (7.24) of Haar measure on G is the unique G-invariant measure on M. Therefore, expectation values of observables $F(\phi)$ in the M model with Hamiltonian $H(\phi)$ are *equal* to expectation values of observables $\tilde{F}(g)\equiv F(g\bar{\phi})$ in the G model with Hamiltonian $\tilde{H}(g)\equiv H(g\bar{\phi})$ [cf. (4.10)–(4.13)]. It suffices, therefore, to simulate the latter model.

Finally, the correctness of the MGMC method for a lattice gauge theory (cf. Sec. V) follows by arguments analogous to those used for the G-valued nonlinear σ model. Indeed, a lattice gauge theory can be thought of (if desired) as a group-valued nonlinear σ model with strange four-spin interactions.

VIII. STOCHASTIC LINEAR ITERATIONS FOR GAUSSIAN MODELS

In this section we analyze an important class of Markov chains, the stochastic linear interactions for Gaussian models.[91] This class includes, among others, the single-site heat-bath algorithm (with deterministic sweep of the sites[92]), the stochastic SOR algorithm,[38,39] and the multigrid Monte Carlo algorithm—all, of course, in the Gaussian case only. We show that the behavior of the stochastic algorithm is completely determined by the behavior of the corresponding deterministic algorithm for solving linear equations.

Consider any quadratic Hamiltonian

$$H(\phi)=\tfrac{1}{2}(\phi,A\phi)-(f,\phi)\ , \tag{8.1}$$

where A is a symmetric positive-definite matrix, and the corresponding Gaussian measure

$$d\pi(\phi)=\text{const}\times e^{-(\phi,A\phi)/2+(f,\phi)}d\phi \tag{8.2}$$

having mean $A^{-1}f$ and covariance matrix A^{-1}. Next consider any first-order stationary linear stochastic iteration of the form

$$\phi^{(n+1)}=M\phi^{(n)}+Nf+Q\xi^{(n)}\ , \tag{8.3}$$

where M, N, and Q are fixed matrices and the $\xi^{(n)}$ are independent Gaussian random vectors with mean zero and covariance matrix C. The iteration (8.3) has a unique stationary distribution if and only if the spectral radius $\rho(M)\equiv\lim_{n\to\infty}\|M^n\|^{1/n}$ is <1; and in this case the stationary distribution is the desired Gaussian measure (8.2) for all f if and only if

$$N=(I-M)A^{-1} \tag{8.4a}$$

and

$$QCQ^T=A^{-1}-MA^{-1}M^T \tag{8.4b}$$

(here a superscript T denotes transpose).[93]

The reader will note the close analogy with the deterministic linear problem (2.2)–(2.4). Indeed, (8.4) is identical with (2.4); and if we take the "zero-temperature limit" in which H is replaced by βH with $\beta \to +\infty$, then the Gaussian measure (8.2) approaches a δ function concentrated at the unique minimum of H (namely, the solution of the linear equation $A\phi = f$), and the "noise" term disappears ($Q \to 0$), so that the stochastic iteration (8.3) turns into the deterministic iteration (2.3). We have the following.

(a) The linear deterministic problem is the zero-temperature limit of the Gaussian stochastic problem; and the first-order stationary linear deterministic iteration is the zero-temperature limit of the first-order stationary linear stochastic iteration. Therefore, any stochastic linear iteration for generating samples from the Gaussian measure (8.2) gives rise to a deterministic linear iteration for solving the linear equation (2.2), simply by setting $Q = 0$.

(b) Conversely, the stochastic problem and iteration are the nonzero-temperature generalizations of the deterministic ones. In principle this means[91] that a deterministic linear iteration for solving (2.2) can be generalized to a stochastic linear iteration for generating samples from (8.2), if and only if the matrix $A^{-1} - MA^{-1}M^T$ is positive semidefinite: just choose a matrix Q satisfying (8.4b) (Ref. 94). In practice, however, such an algorithm is computationally tractable only if the matrix Q has additional nice properties such as sparsity (or triangularity with a sparse inverse).

Example. (1) *Single-site heat-bath (with deterministic sweep of the sites)=stochastic Gaussian.* On each visit to site i, ϕ_i is replaced by a new value ϕ_i' chosen independently from the conditional distribution of (8.2) with $\{\phi_j\}_{j \neq i}$ fixed at their current values: that is, ϕ_i' is Gaussian with mean $(f_i - \sum_{j \neq i} a_{ij}\phi_j)/a_{ii}$ and variance $1/a_{ii}$. When updating ϕ_i at sweep $n+1$, the variables ϕ_j with $j < i$ have already been visited on this sweep, hence have their "new" values $\phi_j^{(n+1)}$, while the variables ϕ_j with $j > i$ have not yet been visited on this sweep, and so have their "old" values $\phi_j^{(n)}$. It follows that

$$\phi_i^{(n+1)} = \left[f_i - \sum_{j<i} a_{ij}\phi_j^{(n+1)} - \sum_{j>i} a_{ij}\phi_j^{(n)} \right] \Big/ a_{ii} + (a_{ii})^{-1/2} \xi_i^{(n)} \tag{8.5}$$

where ξ has covariance matrix I. A little algebra brings this into the matrix form (8.3) with

$$M = -(D+L)^{-1}L^T , \tag{8.6a}$$

$$N = (D+L)^{-1} , \tag{8.6b}$$

$$Q = (D+L)^{-1}D^{1/2} , \tag{8.6c}$$

where D and L are the diagonal and lower-triangular parts of the matrix A, respectively. It is straightforward to verify that (8.4a) and (8.4b) are satisfied.[95] The single-site heat-bath algorithm is clearly the stochastic generalization of the Gauss-Seidel algorithm.

(2) *Stochastic SOR.* For models which are Gaussian (or more generally, "multi-Gaussian"), Adler[38] and Whitmer[39] have shown that the successive over-relaxation (SOR) iteration admits a stochastic generalization: namely,

$$\phi_i^{(n+1)} = (1-\omega)\phi_i^{(n)} + \omega \left[f_i - \sum_{j<i} a_{ij}\phi_j^{(n+1)} - \sum_{j>i} a_{ij}\phi_j^{(n)} \right] \Big/ a_{ii} + \left[\frac{\omega(2-\omega)}{a_{ii}} \right]^{1/2} \xi_i^{(n)} , \tag{8.7}$$

where $0 < \omega < 2$. For $\omega = 1$ this reduces to the single-site heat-bath algorithm. This is easily seen to be of the form (8.3) with

$$M = -(D+\omega L)^{-1}[(\omega-1)D + \omega L^T] , \tag{8.8a}$$

$$N = \omega(D+\omega L)^{-1} , \tag{8.8b}$$

$$Q = [\omega(2-\omega)]^{1/2}(D+\omega L)^{-1}D^{1/2} , \tag{8.8c}$$

where D and L are as before. It is straightforward to verify that (8.4a) and (8.4b) are satisfied.[95]

(3) *Multigrid Monte Carlo (MGMC) method.* The MGMC algorithm (7.11) is identical to the corresponding deterministic MG algorithm (2.12) and (3.1) except that $\mathcal{S}_l$ is a stochastic rather than deterministic updating. Consider, for example, the case in which $\mathcal{S}_l$ is a stochastic *linear* updating (e.g., single-site heat-bath). Then the MGMC is also a stochastic linear updating of the form (8.3): in fact, M equals M_{MG}, the iteration matrix of the corresponding deterministic MG method, and N equals N_{MG}; the matrix Q is rather complicated, but since the MGMC algorithm is correct, Q must satisfy (8.4b). (The easiest way to see that $M = M_{\mathrm{MG}}$ is to imagine what would happen if all the random numbers $\xi^{(n)}$ were zero. Then the stochastic linear updating would reduce to the corresponding deterministic updating, and hence the same would be true for the MGMC updating as a whole.)

(4) *Langevin equation with small time step.* As far as we know, there does not exist any useful stochastic generalization of the Jacobi iteration. However, let us discretize the Langevin equation

$$\frac{d\phi}{dt} = -\tfrac{1}{2}C(A\phi - f) + \xi , \tag{8.9}$$

where ξ is Gaussian white noise with covariance matrix C, using a small time step δ. The result is an iteration of the form (8.3) with

$$M = I - \frac{\delta}{2}CA , \tag{8.10a}$$

$$N = \frac{\delta}{2}C , \tag{8.10b}$$

$$Q = \delta^{1/2}I . \tag{8.10c}$$

This satisfies (8.4a) exactly, but satisfies (8.4b) only up to an error of order δ (Ref. 96). If $C = D^{-1}$, these M, N correspond to a damped Jacobi iteration with $\omega = \delta/2 \ll 1$ (Ref. 97).

It is straightforward to analyze the dynamic behavior of the stochastic linear iteration (8.3). Using (8.3) and (8.4) to express $\phi^{(n)}$ in terms of the independent Gaussian random variables $\phi^{(0)}, \xi^{(0)}, \xi^{(1)}, \ldots, \xi^{(n-1)}$, we find

$$\phi^{(n)} = M^n \phi^{(0)} + \sum_{k=0}^{n-1} M^{n-1-k} N f + \sum_{k=0}^{n-1} M^{n-1-k} Q \xi^{(k)}$$
$$= M^n \phi^{(0)} + (I - M^n) A^{-1} f + \sum_{k=0}^{n-1} M^{n-1-k} Q \xi^{(k)} , \tag{8.11}$$

from which it follows that

$$\langle \phi^{(n)} \rangle = M^n \langle \phi^{(0)} \rangle + (I - M^n) A^{-1} f \tag{8.12}$$

and

$$\mathrm{cov}(\phi^{(s)}, \phi^{(t)}) = M^s \mathrm{cov}(\phi^{(0)}, \phi^{(0)}) (M^T)^t + \sum_{k=0}^{\min(s,t)-1} M^{s-1-k} Q C Q^T (M^T)^{t-1-k}$$
$$= M^s \mathrm{cov}(\phi^{(0)}, \phi^{(0)}) (M^T)^t + \sum_{k=0}^{\min(s,t)-1} M^{s-1-k} (A^{-1} - M A^{-1} M^T) (M^T)^{t-1-k}$$
$$= M^s \mathrm{cov}(\phi^{(0)}, \phi^{(0)}) (M^T)^t + \begin{cases} [A^{-1} - M^s A^{-1} (M^T)^s] (M^T)^{t-s} & \text{if } s \le t , \\ M^{s-t} [A^{-1} - M^t A^{-1} (M^T)^t] & \text{if } s \ge t . \end{cases} \tag{8.13}$$

Now let us either start the stochastic process in equilibrium

$$\langle \phi^{(0)} \rangle = A^{-1} f , \tag{8.14a}$$

$$\mathrm{cov}(\phi^{(0)}, \phi^{(0)}) = A^{-1} , \tag{8.14b}$$

or else let it relax to equilibrium by taking $s, t \to +\infty$ with $s - t$ fixed. Either way, we conclude that in equilibrium (8.3) defines a Gaussian stationary stochastic process with mean $A^{-1} f$ and autocovariance matrix

$$\mathrm{cov}(\phi^{(s)}, \phi^{(t)}) = \begin{cases} A^{-1} (M^T)^{t-s} & \text{if } s \le t , \\ M^{s-t} A^{-1} & \text{if } s \ge t . \end{cases} \tag{8.15}$$

Moreover, since the stochastic process is Gaussian, all higher-order time-dependent correlation functions are determined in terms of the mean and autocovariance. Thus, the matrix M determines the autocorrelation functions of the Monte Carlo algorithm. In particular, the exponential autocorrelation time $\tau_{\exp}$ (slowest decay rate of any autocorrelation function) is given by

$$\exp(-1/\tau_{\exp}) = \rho(M) , \tag{8.16}$$

and this decay rate is achieved by at least one observable which is linear in the field ϕ. In other words, *the convergence rate of the Monte Carlo algorithm is equal to the convergence rate of the corresponding deterministic iteration:* namely, $\rho(M)$.

Similarly we can compute the integrated autocorrelation time $\tau_{\mathrm{int},F}$ for selected observables F. Consider, for example, an observable linear in the field, $F = a \cdot \phi$; we have

$$C_{FF}(t) = (a, M^{|t|} A^{-1} a) \tag{8.17}$$

and hence

$$\tau_{\mathrm{int},F} = \frac{\left(a, \dfrac{I+M}{I-M} A^{-1} a\right)}{(a, A^{-1} a)} . \tag{8.18}$$

In particular, if a is an eigenvector of M^T with eigenvalue λ, then

$$\tau_{\mathrm{int},F} = \frac{1+\lambda}{1-\lambda} . \tag{8.19}$$

Likewise, consider an observable quadratic in the "shifted field" $\tilde{\phi} \equiv \phi - A^{-1} f$: namely, $F = (\tilde{\phi}, K\tilde{\phi})$ for some symmetric matrix K. Then

$$C_{FF}(t) = 2 \,\mathrm{tr}[A^{-1} (M^T)^{|t|} K M^{|t|} A^{-1} K] . \tag{8.20}$$

In particular, if $K = a a^T$ and a is an eigenvector of M^T with real eigenvalue λ, then

$$C_{FF}(t) = 2\lambda^{2|t|} (a, A^{-1} a)^2 \tag{8.21}$$

and, hence,

$$\tau_{\mathrm{int},F} = \frac{1+\lambda^2}{1-\lambda^2} . \tag{8.22}$$

Another way to state these relationships is to recall[98] that the Hilbert space $L^2(\pi)$ is isomorphic to the bosonic Fock space $\mathcal{F}(U)$ built on the "energy Hilbert space" (U, A) defined by (2.37) and (2.38): the "n-particle states" are the homogeneous Wick polynomials of degree n in the shifted field $\tilde{\phi} = \phi - A^{-1} f$. (If U is one-dimensional, these are just the Hermite polynomials.) Then the transition probability $P(\phi^{(n)} \to \phi^{(n+1)})$ induces on the Fock space an operator

$$P = \Gamma(M^T) \equiv I \oplus M^T \oplus (M^T \otimes M^T) \oplus \cdots \tag{8.23}$$

that is the second quantization[98] of the operator M^T on the energy Hilbert space. To see this, let us first rewrite the transition probability $P(\phi \to \phi')$ as an explicit integral kernel:

$$P(\phi \to \phi') = [\det(2\pi Q C Q^T)]^{-1/2} \exp[-\tfrac{1}{2}(\phi' - M\phi - Nf, (QCQ^T)^{-1} (\phi' - M\phi - Nf))] d\phi'$$
$$= [\det(2\pi Q C Q^T)]^{-1/2} \exp[-\tfrac{1}{2}(\tilde{\phi}' - M\tilde{\phi}, (QCQ^T)^{-1} (\tilde{\phi}' - M\tilde{\phi}))] d\tilde{\phi}' . \tag{8.24}$$

(To lighten the notation, let us henceforth drop the tildes.) Now let us apply this integral kernel, as in (6.10), to the Wick exponential[98]

$$g_a(\phi) = :\exp(a\cdot\phi): = \exp(a\cdot\phi - \tfrac{1}{2}a^T A^{-1} a) \ . \tag{8.25}$$

We obtain

$$\begin{aligned}(Pg_a)(\phi) &= [\det(2\pi QCQ^T)]^{-1/2} \exp(-\tfrac{1}{2}a^T A^{-1}a) \int \exp[-\tfrac{1}{2}(\phi' - M\phi, (QCQ^T)^{-1}(\phi' - M\phi)) + a\cdot\phi'] d\phi' \\ &= \exp[-\tfrac{1}{2}(a, MA^{-1}M^T a) + (M^T a, \phi)] = :\exp(M^T a\cdot\phi): \ .\end{aligned} \tag{8.26}$$

On the other hand, by definition,

$$\begin{aligned}\Gamma(M^T):\exp(a\cdot\phi): &= \Gamma(M^T) \sum_{k=0}^{\infty} \frac{1}{k!} :(a\cdot\phi)^k: \\ &= \sum_{k=0}^{\infty} \frac{1}{k!} :(M^T a\cdot\phi)^k: \\ &= :\exp(M^T a\cdot\phi): \ .\end{aligned} \tag{8.27}$$

We have therefore demonstrated the validity of (8.23) on Wick exponentials (8.25). But linear combinations of the Wick exponentials form a dense set in the Hilbert space $L^2(\pi)$, so (8.23) holds universally.[99]

It follows from (8.23) that

$$\| \Gamma(M)^n \restriction \mathbf{1}^\perp \|_{L^2(\pi)} = \| M^n \|_{(U,A)} \tag{8.28}$$

and hence that

$$\rho(\Gamma(M) \restriction \mathbf{1}^\perp) = \rho(M) \ . \tag{8.29}$$

Moreover, P is self-adjoint on $L^2(\pi)$ (i.e., satisfies detailed balance) if and only if M is self-adjoint with respect to the energy inner product (2.37), i.e., if[97]

$$MA = AM^T \ ; \tag{8.30}$$

and in this case

$$\begin{aligned}\rho(\Gamma(M) \restriction \mathbf{1}^\perp) &= \| \Gamma(M) \restriction \mathbf{1}^\perp \|_{L^2(\pi)} \\ &= \rho(M) = \| M \|_{(U,A)} \ .\end{aligned} \tag{8.31}$$

In summary, we have shown that the dynamic behavior of any stochastic linear iteration is completely determined by the behavior of the corresponding deterministic linear iteration. In particular, for Gaussian MGMC, the results of Ref. 63, combined with the arguments of the present section, prove rigorously that *critical slowing down is completely eliminated.* That is, the autocorrelation time τ of the MGMC method is *bounded* as criticality is approached (empirically $\tau \approx 1-2$).

Likewise, we can analyze exactly the Gaussian stochastic SOR algorithm of Adler[38] and Whitmer.[39] Consider, for example, the massless free field with Dirichlet boundary conditions on a square $\{1,\ldots,L\}\times\{1,\ldots,L\}$. Then it is well known[100] that the SOR iteration with overrelaxation parameter ω (with red-black or lexicographic ordering) has convergence rate

$$\rho(M_{\mathrm{SOR},\omega}) = \begin{cases} \dfrac{\omega^2\mu^2 - 2(\omega-1) + \omega\mu\sqrt{\omega^2\mu^2 - 4(\omega-1)}}{2} & \text{if } \omega \le \omega_b \ , \\ \omega - 1 \quad \text{if } \omega \ge \omega_b \ , \end{cases} \tag{8.32}$$

where

$$\mu = \cos\frac{\pi}{L+1} \tag{8.33}$$

and

$$\omega_b = \frac{1}{1+\sqrt{1-\mu^2}} = \frac{2}{1+\sin\dfrac{\pi}{L+1}} \tag{8.34}$$

Therefore, $\omega_{\mathrm{optimal}} = \omega_b$, and

$$\rho(M_{\mathrm{SOR},\omega_{\mathrm{optimal}}}) = \frac{1-\sin\dfrac{\pi}{L+1}}{1+\sin\dfrac{\pi}{L+1}} \approx 1 - \frac{2\pi}{L} \ . \tag{8.35}$$

It follows that, for the stochastic SOR algorithm, $\tau_{\exp}(\omega_{\mathrm{optimal}}) \sim L$ and the dynamic critical exponent is $z = 1$. Analogous results can easily be derived for other boundary conditions.

IX. PERFORMANCE OF THE MULTIGRID MONTE CARLO ALGORITHM FOR NON-GAUSSIAN MODELS

It is pleasant to know that the MGMC method eliminates critical slowing down for Gaussian models (free fields), but the real test of any algorithm is its performance on interacting (non-Gaussian) field theories. It turns out that the performance of MGMC on non-Gaussian models depends strongly on the specific form of the nonlinearities. We have therefore undertaken systematic numerical experiments[4-7] on a variety of models, with the aim of refining our *physical insight* into *why* MGMC works well on some models and not so well on others. In this section we summarize the currently available heuristic and numerical evidence on the performance of MGMC for non-Gaussian models, and explain what work needs to be done in the future.

A. ϕ^4 models

The first numerical experiments[4] on non-Gaussian MGMC concerned the ϕ^4 model

$$H(\phi)=\frac{\alpha}{2}\sum_{|x-x'|=1}(\phi_x-\phi_{x'})^2+\sum_x(\lambda\phi_x^4+A\phi_x^2) \tag{9.1}$$

in dimension $d=2$. We ran on a 128×128 periodic lattice at $\alpha=1$, $\lambda=0.1$ at a variety of values of A (bare mass squared) near the critical point $A_c\approx-0.3$. Our MGMC interaction used one heat-bath presweep and one heat-bath postsweep ($m_1=m_2=1$) with red-black ordering, and a W cycle ($\gamma=2$); for comparison, we also ran a pure heat-bath algorithm ($m_1=m_2=1$, $\gamma=0$). We measured the integrated autocorrelation time (6.5) for the total magnetization $\mathcal{M}\equiv\sum_x\phi_x$, and found that the MGMC algorithm does *not* eliminate critical slowing down for this model. Indeed, we found that the autocorrelation time for MGMC diverges at the critical point with the *same* dynamic critical exponent as in the heat-bath algorithm; the reduction in τ over the heat-bath algorithm (at this value of λ) is a factor of ≈20. Since each MGMC iteration takes twice the work of the corresponding heat-bath iteration,[101] the gain in efficiency, measured in CPU units, is a factor of ≈10. Such a gain is perhaps significant, but it is disappointing that the dynamic critical exponent was not reduced at all.

In retrospect, it is now clear that the ϕ^4 model was one of the *worst* possible choices for a test of the MGMC method. In fact, we can argue heuristically that the efficiency gain near criticality in this model should approach a constant factor $F(\lambda)<\infty$, where $F(\lambda)$ is a decreasing function of λ. Of course, $F(\lambda)$ must approach $+\infty$ as $\lambda\to0$, in accordance with the absence of critical slowing down for the Gaussian MGMC method. Here is our heuristic analysis of the behavior of the MGMC method for ϕ^4 models.

Let us use the "unigrid" point of view (Sec. II C): this means that we imagine performing updates successively on single sites, cubes of side 2, cubes of side 4, cubes of side 8, and so on; an update of a block B consists of proposing to add a constant t simultaneously to all the spins in that block, computing the conditional Hamiltonian $H_{\rm cond}(t)=H(\phi+t\chi_B)$, and choosing a random value of t according to the probability density $P_{\rm cond}(t)\sim\exp[-H_{\rm cond}(t)]$.

Consider first the Ising model, which is the limiting case $\lambda\to\infty$ with $A=-2\lambda$; here the only allowed spin values are $+1$ and -1. What does the probability distribution $P_{\rm cond}(t)$ look like? If *all* of the spins of the block B happen to be in the state -1, then the possible values of t (i.e., those that have nonzero probability) are 0 and $+2$; and analogously if all the spins in B are $+1$; but in all other cases, the only allowed value for t is *zero*. In other words, the MGMC updates *do nothing*, except in the rare event that all of the spins in B are initially aligned. Since the probability of such a configuration decreases exponentially with the volume of the block B—indeed, it is already quite improbable even for a 2×2 block—it follows that the MGMC method is completely ineffective on long length scales, and should behave essentially the same as a single-site heat-bath algorithm.

The same is clearly true when λ is large but finite: The potential $V(\phi)$ has deep double wells at $\phi=\pm c$, so that every spin B has a value very near either $+c$ or $-c$. Then, unless all the spins in B happen to be in the *same* well, the only probable values for t are *very near zero,* so that the MGMC updates induce only small fluctuations *within* the current well. These are, to be sure, *long-wavelength* small fluctuations; but intuition tells us that the important large-scale collective modes are *not* small fluctuations, but are rather Ising type (i.e., flips in the *sign* of ϕ), so the MGMC updates are again ineffective in moving the system around the important regions of the configuration space.

Consider, finally, the general case ($\lambda>0$ of arbitrary magnitude). Then the conditional Hamiltonian for a block B of size b^d is

$$H_{\rm cond}(t)=\lambda' t^4+\kappa' t^3+A' t^2+h' t+\text{const}\ , \tag{9.2a}$$

where

$$\begin{aligned}\lambda'&=\sum_{x\in B}\lambda=b^d\lambda\ ,\\ \kappa'&=\sum_{x\in B}4\lambda\phi_x\ ,\\ A'&=\sum_{x\in B}(6\lambda\phi_x^2+A)+db^{d-1}\alpha\ ,\\ h'&=\sum_{x\in B}(4\lambda\phi_x^3+2A\phi_x)\ .\end{aligned} \tag{9.2b}$$

Now suppose that

$$a\equiv\min_{-\infty<t<\infty}\frac{d^2}{dt^2}H_{\rm cond}(t)>0\ . \tag{9.3}$$

Then $H_{\rm cond}$ is strictly convex, hence has a unique local minimum (call it t_0), and

$$H_{\rm cond}(t)\ge H_{\rm cond}(t_0)+\frac{a}{2}(t-t_0)^2\ . \tag{9.4}$$

One expects heuristically, therefore, that "typical" values of the random variable t in the probability distribution $P_{\rm cond}(t)\sim\exp[-H_{\rm cond}(t)]$ will differ from t_0 by at most $\approx a^{-1/2}$. [In fact, since

$$H_{\rm cond}(t)\ge H_1(t)+\frac{a}{2}(t-t_0)^2 \tag{9.5}$$

with H_1 convex, the Brascamp-Lieb inequality[102] implies rigorously that $\mathrm{var}(t)\le a^{-1}$.] Now a simple computation shows that

$$a=2A+2db^{d-1}\alpha+12\lambda b^d\left[\frac{1}{b^d}\sum_{x\in B}(\phi_x-\bar\phi_x)^2\right]\ , \tag{9.6a}$$

where

$$\bar\phi_x\equiv\frac{1}{b^d}\sum_{x\in B}\phi_x\ . \tag{9.6b}$$

For a one-site block, the term in large parentheses vanishes (as it should); but for a large block, the term in large

parentheses is with very high probability close to

$$\mathrm{var}(\phi_x) \equiv \langle \phi_x^2 \rangle - \langle \phi_x \rangle^2 = \langle \phi_x^2 \rangle > 0 \tag{9.7}$$

(Ref. 103). On the other hand, $\langle \phi_x^2 \rangle$ in the interacting theory is at least as great as its value $\langle \phi_x^2 \rangle_0$ in the theory with the ferromagnetic nearest-neighbor interaction turned off, namely in the single-spin measure

$$d\nu(\phi) \sim \exp[-\lambda\phi^4 - (A + d\alpha)\phi^2]\, d\phi \tag{9.8}$$

(this is Griffiths' second inequality[104]). It follows[105] that

$$A + d\alpha \geq -2\lambda \langle \phi_x^2 \rangle_0 \geq -2\lambda \langle \phi_x^2 \rangle \ . \tag{9.9}$$

Therefore, for large blocks ($b \gg 1$) we have

$$a \gtrsim 2db^{d-1}\alpha + 12\lambda b^d \langle \phi_x^2 \rangle \tag{9.10}$$

and hence

$$\mathrm{var}(t) \lesssim (2db^{d-1}\alpha + 12\lambda b^d \langle \phi_x^2 \rangle)^{-1} \ . \tag{9.11}$$

Now the term $2db^{d-1}\alpha$ is present already for free fields (it measures the energy cost of making a piecewise-constant update of a Gaussian field); from the rigorous convergence theorem for the Gaussian MGMC algorithm,[47] we know that an update of this size is "big enough" to eliminate critical slowing-down, provided that $\lambda \geq 2$ (i.e., a W cycle is used). On the other hand, we might expect that if the size of the update on a given length scale is drastically less than this, the MGMC updates are essentially *ineffective* on that length scale.[106] We conclude that for block sizes

$$b \gtrsim \alpha/\lambda \langle \phi_x^2 \rangle \sim 1/\lambda \ , \tag{9.12}$$

the MGMC updates are ineffective. The scenario is therefore the following: the first few coarse grids (up to $b \approx 1/\lambda$) do useful work, while the rest are useless, so that the "effective correlation length" of the theory is reduced by a factor of about $1/\lambda$; we expect, therefore, that the autocorrelation time of the MGMC method will be a factor of about $(1/\lambda)^z$ smaller than the autocorrelation time of the heat-bath algorithm (here $z \approx 2$ is the dynamic critical exponent). This substantiates the claims made above, and makes the specific prediction that the improvement factor $F(\lambda)$ behaves as

$$F(\lambda) \sim 1/\lambda^z \tag{9.13}$$

as $\lambda \to 0$.

The foregoing arguments have, in our opinion, too many loopholes to justify taking seriously the quantitative prediction (9.13). Nevertheless, we are convinced that the qualitative behavior is correct: for one-component ϕ^4 models, the MGMC method has the *same* dynamic critical exponent as the heat-bath algorithm; and the improvement factor $F(\lambda)$ is a decreasing function of λ which approaches $+\infty$ as $\lambda \to 0$. It would be an interesting project to measure carefully the improvement factor $F(\lambda)$ along with the histogram of update amplitudes $\mathrm{var}(t)$ on the various course grids, so as to test quantitatively the reasoning (9.2)–(9.13).

We emphasize that this argument applies to ϕ^4 models in *any lattice dimension:* the argument has nothing whatsoever to do with the behavior of the renormalization-group flow (which would be different for $d < 4$ and $d > 4$). This illustrates, once again, that the "flow" of conditional Hamiltonians in the MGMC method is *very different* from the flow of marginal (renormalized) Hamiltonians in the block-spin renormalization group.

We suspect, in fact, that the MGMC method is doomed to behave similarly on any lattice model with a *nonconvex* Hamiltonian.[107] On the other hand, it is at least conceivable that the MGMC method might eliminate critical slowing down for all models (in some large class) with a *convex* Hamiltonian. (This is admittedly a wild speculation.) One interesting model in which the critical point occurs when the Hamiltonian is still convex (albeit barely so) is the anharmonic crystal[108]

$$H(\phi) = \sum_{|x - x'| = 1} V(\phi_x - \phi_{x'}) \tag{9.14}$$

where V is, for example, a polynomial of degree 4. (Note that the coarse-grid Hamiltonian is again of this form, but with a space-dependent V_{coarse}; and if V is convex, then so is V_{coarse}.) It would be interesting to try the MGMC method on this model.

We remark, finally, that although the *additive* MGMC algorithm (Secs. III and VII) is likely to perform badly for N-component ϕ^4 models for all N, we expect that the *multiplicative* MGMC algorithm (Secs. IV and VII) will behave well for N-component ϕ^4 models ($N \geq 2$) whenever it behaves well for the corresponding $O(N)$-symmetric nonlinear σ model. This is based on the intuition that the important large-scale collective physics of a multicomponent ϕ^4 model is contained in the *angular* variables; we expect that the *magnitudes* of ϕ, conditional on the angles, can be equilibrated well by a single-site heat bath. This prediction requires, of course, a careful numerical check.

B. Two-dimensional XY model

We are currently completing a comprehensive study of MGMC for the two-dimensional XY (plane-rotator) model.[5] We ran on lattice sizes up to 128×128 at a series of temperatures very near the critical temperature $\beta_c \approx 1.13$, and carried out a finite-size-scaling analysis of both static and dynamic quantities. The results for the MGMC method (W cycle) are roughly as follows: As the critical temperature is approached from above, τ appears to diverge, with approximately the same dynamic critical exponent as in the heat-bath algorithm; the reduction in τ over the heat-bath algorithm is a factor of ≈ 20, yielding an efficiency gain of a factor of ≈ 10. On the other hand, below the critical temperature, τ is very small ($\approx 1-2$; since for the heat-bath algorithm τ is unbounded (as $L \to \infty$) in this regime, the gain in efficiency is unbounded as well. The finite-size scaling analysis shows a very subtle crossover between these two regimes.

This behavior can be understood physically: in the low-temperature phase the main excitations are spin waves, which are well handled by MGMC (as in the Gaussian model); but near the critical temperature the important excitations are widely separated vortex-

antivortex pairs,[109] which are apparently not so easily created by the MGMC updates.

Analogous behavior is to be expected in the four-dimensional U(1) lattice gauge theory.

C. Asymptotically free theories

The simplest asymptotically free theories are the $O(N)$-symmetric nonlinear σ models ($N>2$) in two dimensions. These models can be handled by multiplicative MGMC as discussed in Secs. IV and VII. In particular, the O(4) model is isomorphic to the SU(2) principal chiral model, which simplifies the technical details of the simulation. We have recently begun a test of the MGMC method for the two-dimensional O(4) model.[6] Very preliminary results show a large reduction in critical slowing down—possibly its complete elimination. For example, on a 128×128 lattice at $\beta=2.828$, we find $\tau_{\text{heat bath}}\approx300$ compared to $\tau_{\text{MGMC}}\approx10$; and τ_{MGMC} appears to *decrease* as β is increased, contrary to the behavior of $\tau_{\text{heat bath}}$. Comprehensive tests are now in progress.

We can, in fact, argue quite generally that for asymptotically free theories with a continuous symmetry group (e.g., nonlinear σ models or non-Abelian lattice gauge theories), the multiplicative MGMC algorithm (with a W cycle) as described in Secs. IV, V, and VII should *eliminate entirely* the critical slowing down except for a possible logarithm. Recall our argument for ϕ^4 models: the first few coarse grids (up to some scale $b_{\max}$) do useful work, but the rest are ineffective; therefore, the "effective correlation length" is reduced from ξ to $\approx\xi/b_{\max}$, and the autocorrelation time is reduced correspondingly. For asymptotically free theories with a continuous symmetry group, on the other hand, the important excitations at short wavelengths are weakly-interacting spin waves;[110] and this "almost-Gaussian" behavior persists up to length scales around $\approx\xi/10$, where ξ is the correlation length (inverse particle mass). (Of course, at length scales of order ξ the behavior is far from Gaussian, giving rise to scattering, bound states, and other nontrivial physics.) Since the physics on length scales up to $\approx\xi/10$ is approximately Gaussian, one might expect that the MGMC updates (using a W cycle) are essentially perfect on these length scales; but the MGMC updates on yet coarser grids are likely to be ineffective, because they fail to take into account the relevant nonlinear collective excitations. Therefore, the "effective correlation length" is reduced from ξ to $\approx\xi/(\xi/10)\approx10$—that is, a *bounded* quantity—so the autocorrelation time should correspondingly be bounded. Of course, it is not quite true that the short-wavelength fluctuations are perfectly Gaussian—in fact, the coupling strength on scale b behaves like $1/\log^p(\xi/b)$ for some power $p>0$—so it would not be surprising if τ contained a logarithm, i.e., $\tau_{\text{MGMC}}\sim\log^q\xi$ for some power $q>0$. But we expect that the critical slowing down will not be worse than this (for a W cycle).

Note the importance of the qualifying words "with a continuous symmetry group" and "*multiplicative*" MGMC (Secs. IV, V, and VII) in the foregoing argument. Consider, for example, the ϕ^4 model in dimension $d>4$ (with any number of components). This model is infrared asymptotically free, in the sense that the behavior on long length scales of RG-type block spins is essentially Gaussian; but, as argued above, we expect that the MGMC method (in the *additive* style described in Secs. III and VII) does *not* reduce the dynamic critical exponent. The point is that almost Gaussianness in the RG sense in *not* equivalent to almost Gaussianness in the MGMC sense. Nevertheless, our intuitive understanding of the physics of nonlinear σ models and non-Abelian lattice gauge theories leads us to believe that these models are probably almost Gaussian in *both* senses.

These heuristic arguments obviously require a careful numerical test, first for two-dimensional σ models and then (hopefully) for four-dimensional non-Abelian gauge theories.

D. Discrete models

The multiplicative MGMC algorithm described in Secs. IV, V, and VII is in principle applicable to σ models and lattice gauge theories based on *any* compact group, whether continuous or discrete. But though the algorithm will work correctly, it will not necessarily work well. Indeed, the fundamental physical idea underlying the MGMC method is that the system can be equilibrated by *large-scale, small-amplitude* updates of fixed shape. More precisely, one offers (in the unigrid interpretation) an update $\phi\to\phi+t\chi_B$ (or $g\to t\circ_B g$) and lets the system choose the update amplitude t according to the probability distribution $P_{\text{cond}}(t)\sim\exp[-H(\phi+t\chi_B)]$. But this amplitude turns out to be small if the block size is large—e.g., of order $b^{-(d-1)/2}$ in the Gaussian case. On the other hand, for a discrete model, the only allowable "small" update amplitude is *zero*. So the MGMC moves on large blocks do almost nothing.

For example, for the Ising model ($=Z_2$ principal chiral model), a multiplicative MGMC update (in the unigrid interpretation) consists of proposing to flip all the spins in a block B. But in equilibrium, the majority of the "boundary bonds" $\langle ij\rangle$ with $i\in B$, $j\notin B$ are "satisfied" ($\sigma_i\sigma_j=+1$), so that flipping the spins in B typically costs an energy of order b^{d-1} (the surface area of B). If the block B is large, such a proposal would almost always be rejected. Therefore, only the first few coarse grids would do useful work, and the autocorrelation time of the MGMC algorithm would be only slightly better than for the single-site heat-bath-algorithm.

More generally, we suspect that collective-mode updates of *fixed shape* are doomed to failure in discrete-spin models. Rather, it appears necessary to devise some (model-dependent) way to *let the system choose its collective modes* (e.g., the shape of blocks to be flipped). In general it is very difficult to do this and still preserve the correct Gibbs measure; but Swendsen and Wang[17] and others[18–25] have recently made great progress in this direction (see Sec. X C).

X. DISCUSSION

In this section we would like to comment on some alternative MGMC schemes proposed recently by other

workers, and to compare the behavior of MGMC to other "collective-mode" algorithms such as Fourier acceleration[11,12] and the Swendsen-Wang approach.[17–25]

One difficulty in comparing Monte Carlo algorithms is the paucity of good "experimental" data. Some papers propose a new method without actually trying it. More common—and in our opinion, even more vexing—are papers that present *some* numerical data, but not enough to confirm the sweeping claims made for the method. From a theoretical point of view, any Monte Carlo method is itself a stochastic dynamical system and has its own autocorrelation-time critical exponent z: $\tau \sim \xi^z$ where ξ is the correlation length. All the methods for estimating physical critical exponents (finite-size scaling and possibly even Monte Carlo renormalization group) could be used to estimate z (Ref. 111).

A. Other approaches to the multigrid Monte Carlo method

The idea of using multigrid techniques to update the (stochastic) values of a Euclidean quantum field was apparently first proposed by Parisi[112] in 1983, in the same paper in which he proposed Fourier acceleration. However, his brief description of a multigrid Monte Carlo algorithm was not quite correct. He proposed to write the field ϕ as a superposition of contributions from each length scale,

$$\phi_x = \sum_{l=0}^{M} \sum_{y \in \Omega_l} c_{x,y}^{(l)} \phi_y^{(l)} , \tag{10.1}$$

where the $c_{x,y}^{(l)}$ are suitable interpolation weights. But this formulation is correct—in the sense that it gives rise to the correct Gibbs measure—only if the fields $\phi^{(l)}$ are constrained to be "orthogonal" in an appropriate sense. [For example, if one were to use piecewise-constant interpolation, then each field $\phi^{(l)}$ (except for the coarsest-grid field $\phi^{(0)}$) should have zero mean on each basic cube of side 2, so as to be "orthogonal" to the next-coarser field $\phi^{(l-1)}$.] Such a decomposition *with constraints* appears, for example, in the rigorous work of Gallavotti and collaborators[113] and Gawedzki and Kubiainen.[114]

A correct version of the MGMC algorithm based on the decomposition-with-constraints idea was proposed by Mack[115] in 1987. In the most straightforward implementation, one would update successively the fields $\phi^{(l)}$, $0 \le l \le M$, using a heat-bath algorithm that is specially designed to respect the constraints. However, this algorithm is notably awkward to implement, as a result of the constraints. Mack advocates, therefore, an alternative algorithm in which the fields

$$\Phi_x^{(l)} \equiv \sum_{l'=0}^{l} \sum_{y \in \Omega_{l'}} c_{x,y}^{(l,l')} \phi_y^{(l')} , \tag{10.2}$$

which are essentially the sum of $\phi^{(l)}$ *and all coarser-grid fields,* are updated using an ordinary heat-bath (or Metropolis) algorithm, with the finer-grid fields $\phi^{(M)}$, $\phi^{(M-1)}, \ldots, \phi^{(l+1)}$ held fixed. But in this form, it seems to us that Mack's MGMC algorithm is essentially *equivalent* to ours, with the only difference being the choice of interpolation operator (we use piecewise-constant, while he uses a more complicated but smoother smearing) and the fact that Mack uses a "unigrid" implementation (Sec. II C). We are therefore unable to understand Mack's claim that his MGMC algorithm is more effective than ours in beating critical slowing down. In the free-field case, piecewise-linear interpolation does indeed perform better than piecewise-constant, but for the W cycle ($\gamma \ge 2$) the difference is only a constant factor; that is the content of the rigorous convergence theorem.[47,116] On the other hand, for ϕ^4 models both versions of the MGMC method should perform equally badly, as our argument in Sec. IX A shows [the λb^d term in (9.11) will be present irrespective of the choice of interpolation]. Indeed, Mack has a rather similar understanding of the reasons for possible failure of the MGMC method in non-Gaussian models, as a result of strong coupling between grids arising from the nonquadriatic terms in the Hamiltonian.[117] Unfortunately, Mack's paper[115] contains only very preliminary numerical results; a promised future paper[118] will hopefully contain detailed measurements of critical slowing down (including a finite-size-scaling analysis) for MGMC versus single-site heat-bath-methods.

A *unigrid* algorithm essentially equivalent to our multiplicative MGMC algorithm for nonlinear σ models (Secs. IV and VII) was proposed in 1985 by Meyer-Ortmanns.[119] The only difference is that she chooses the block size (and location) randomly, while we choose them periodically in a fixed sequence (see Sec. II C). However, provided that the probability distribution of block sizes gives sufficient weight to large blocks, we expect that this algorithm will behave essentially the same as ours. Her algorithm suffers, of course, from the disadvantage shared by all unigrid algorithms: the work per iteration grows faster than the volume [cf. (2.42)]. Nevertheless, we feel that Meyer-Ortmanns deserves credit for identifying *physically* the collective modes which should be updated.

An MGMC algorithm very similar to our own was proposed independently by Brandt, Ron, and Amit.[120] The only difference between their algorithm and ours is the choice of coarse-grid Hamiltonian: we adhere strictly to the variational definition $H_{l-1}(\psi) \equiv H_l(\phi + p_{l,l-1}\psi)$, while Brandt, Ron, and Amit assert the freedom to choose freely the coarse-grid Hamiltonian, as in the deterministic MG method.[121] It seems to us that their assertion is incorrect, and that their algorithm does *not in general leave invariant the correct Gibbs measure.*

The paper of Brandt, Ron, and Amit, does, however, contain an interesting point of view on highly nonlinear field theories. They argue[122] that the "contour map" of the Hamiltonian H typically has many local minima (or "basins"), such that transitions between these basins are difficult if one uses local (e.g., single-site) updates. The role of the collective-mode (e.g., coarse-grid) updates is precisely to cause such transitions. Unfortunately, they point out, a collective-mode update is not likely to land us near the bottom of the new basin; rather, it is most likely to land us high on the walls of the new basin, at an energy level much higher than the current energy near the bottom of the old basin. Therefore, such a move

(considered as a Metropolis proposal) will almost certainly be rejected. On the other hand, if we could explore the new basin, using finer-scale moves to arrive near its bottom, and *then* decide to accept or reject the entire package of moves, the transition might have a reasonable chance of being accepted. Unfortunately, while this idea may well be useful in *minimization* (zero-temperature) problems, where "anything is fair," we have been unable to see how to apply it to *finite-temperature* problems, where the requirement of leaving invariant the Gibbs measure imposes severe constraints on acceptable algorithms.[123]

Other "multigrid Monte Carlo" algorithms have been proposed by Gupta[124] and by Decker.[125] We do not fully understand these algorithms, but they appear to be incorrect, in the sense that they fail to leave invariant the correct Gibbs measure. Both algorithms involve "(approximate) renormalized Hamiltonians" (in the RG sense) and "tunable parameters." We emphasize once again that (1) multigrid Monte Carlo is very different from the renormalization-group method. In particular, the coarse-grid Hamiltonian in the MGMC method is neither equal nor approximately equal to the renormalized Hamiltonian in the RG method; and (2) as far as we know, *there is no freedom in choosing the coarse-grid Hamiltonian:* it *must* be defined by the variational formula $H_{l-1}(\psi) \equiv H_l(\phi + p_{l,l-1}\psi)$. We know of no other way to leave invariant the correct Gibbs measure.

Finally, Schmidt[126] and Brower, Giles, Moriarty, and Tamayo[127] have proposed a class of Monte Carlo algorithms based on simulating on exact or approximate renormalized Hamiltonian in the RG sense. When an exact renormalized Hamiltonian is used, these algorithms become *static* Monte Carlo algorithms (Sec. VI); they are stochastic analogues of the "total reduction"[128] (and "nested dissection"[129]) algorithms for solving partial differential equations (PDE's). Unfortunately, an exact renormalized Hamiltonian can be used only in trivial (exactly soluble) one-dimensional problems; and in all other cases, the efficiency of the algorithm appears to deteriorate *exponentially* in the lattice volume (i.e., even worse than the critical slowing down of conventional algorithms).

B. Comparison with Fourier acceleration

The "collective-mode" algorithm closest in spirit to MGMC is Fourier acceleration.[11,12] Both algorithms are inspired by free-field theory, and both use predefined additive collective modes. These modes are piecewise constant (or piecewise-linear) in the MGMC method while they are sine waves in Fourier acceleration; but the idea is fundamentally the same. The behavior of the two algorithms in the free-field case is nearly identical; both algorithms reduce the autocorrelation time to a small constant (i.e., a few sweeps), independent of the correlation length and lattice size.

At a deeper level, we suspect that the MGMC method and Fourier acceleration have the same qualitative behavior also for *non-Gaussian* field theories, in the sense that they probably work well for the same models and work badly for the same models. For example, Fourier acceleration is just as unlikely to promote spin flips in a ϕ^4 model as is the MGMC method. [In fact, it is *less* likely: as a small-step-size algorithm, energy barriers must be climbed over slowly, not jumped over. This difficulty has nothing to do with *critical* slowing-down: it would occur even in a *single-site* ϕ^4 model in the deep double-well regime, where a small-step-size algorithm (e.g., Langevin, microcanonical, or hybrid) would perform vastly worse than a single-site heat-bath algorithm.] Likewise, in the two-dimensional XY model, Fourier acceleration should behave well on the low-temperature side of criticality, where the relevant excitations are spin waves, but not on the high-temperature side of criticality, where the relevant excitations are vortex-antivortex pairs. Indeed, the fragmentary available data[130] appear to show such a behavior. Finally, for the SU(3) principal chiral model in two dimensions—which is asymptotically free—an initial test[131] of the Fourier-accelerated Langevin and hybrid algorithms suggested a significant reduction in critical slowing down (possibly its complete elimination).

The choice between the MGMC method and Fourier acceleration will likely come down, therefore, to a question of the constant prefactors. Such a question can be decided only by careful measurements of the autocorrelation time (and CPU time) in the MGMC method versus the Fourier accelerated Langevin[11] and hybrid[12] algorithms. (We emphasize that a fair test must use the *exact* versions of the Langevin[132] and hybrid[133] algorithms, and should tune the time-step size in these algorithms so as to optimize the autocorrelation time.) We suspect that the MGMC method will turn out to be slightly more efficient, because it is not restricted to "small" steps but this may depend on the details of the model, and in particular on the efficiency of the heat-bath subroutine.

Fourier acceleration does, however, have at present one *major* advantage: it is applicable to non-Abelian gauge theories and to models with dynamical fermions. It remains to be seen where the MGMC method can be generalized (in a practical way) to these models.

C. Other collective-model algorithms

A very different type of collective-mode algorithm was proposed recently by Swendsen and Wang[17] for Potts models, and subsequently generalized in different directions by several workers.[18–25] Basically, the idea is to augment the given model by means of *auxiliary variables,* and then to simulate this augmented model.

The Swendsen-Wang (SW) algorithm for the ferromagnetic Potts model

$$d\mu_{\text{Potts}}(\sigma) = Z^{-1} \exp\left[\sum_{\langle ij \rangle} J_{ij}(\delta_{\sigma_i,\sigma_j} - 1)\right] d\mu_0(\sigma) \tag{10.3a}$$

$$= Z_{\text{Potts}}^{-1} \prod_{\langle ij \rangle} [(1-p_{ij}) + p_{ij}\delta_{\sigma_i,\sigma_j}] d\mu_0(\sigma) \tag{10.3b}$$

[where $p_{ij} \equiv 1 - \exp(-J_{ij})$] is based on simulating the augmented model

$$d\mu(\sigma,n)=Z^{-1}\prod_{\langle ij\rangle}[(1-p_{ij})\delta_{n_{ij},0}+p_{ij}\delta_{n_{ij},1}\delta_{\sigma_i,\sigma_j}] \times d\mu_0(\sigma)d\mu_0(n)\ , \tag{10.4}$$

which has q-state Potts spins σ_i at the sites and occupation variables $n_{ij}=0,1$ on the bonds. It is easily seen that integrating over the bond variables $\{n\}$ in (10.4) yields (10.3). Moreover, the augmented model (10.4) has very simple conditional probability distributions.

(1) *Distribution of* $\{n\}$ *given* $\{\sigma\}$. Independently for each bond $\langle ij\rangle$, one sets $n_{ij}=0$ in case $\sigma_i\neq\sigma_j$, and sets $n_{ij}=0,1$ with probability $1-p_{ij},p_{ij}$, respectively, in case $\sigma_i=\sigma_j$.

(2) *Distribution of* $\{\sigma\}$ *given* $\{n\}$. Independently for each connected cluster of sites (in the graph whose edges are the bonds having $n_{ij}=1$), one sets all the spins σ_i in the cluster to the same value, chosen equiprobably from the set $\{1,\ldots,q\}$.

The Swendsen-Wang algorithm them simulates (10.4) by alternately applying these two conditional distributions—that is, by alternately generating new bond occupation variables (independent of the old ones) given the spins, and new spin variables (independent of the old ones) given the bonds.[134]

It is certainly plausible that the SW algorithm might have less critical slowing down than the conventional algorithms: the reason is that a local move in one set of variables can have highly nonlocal effects in the other. For example, setting $n_{ij}=0$ on a single bond may disconnect a cluster, causing a big subset of the spins in that cluster to be flipped simultaneously. In some sense, therefore, the SW algorithm is a collective-mode algorithm in which the collective modes are *chosen by the system* rather than imposed from the outside as in multigrid or Fourier acceleration.

In fact, the performance of the SW algorithm for ferromagnetic Potts models (at *second*-order transitions) is nothing short of extraordinary. For the two-dimensional Ising model at the bulk critical temperature, preliminary data[135] on lattices of size $L=64,128,256,512$ show that the autocorrelation time $\tau_{\text{int},\mathcal{E}}$ ($\mathcal{E}$=energy≈slowest mode) grows from ≈ 5 at $L=64$ to only ≈ 8 at $L=512$. This is consistent with a dynamic critical exponent $z\approx 0.3$ (Ref. 17); but *precisely because* τ rises so slowly with L, it is difficult to estimate z with much precision. For the two-dimensional three-state Potts model, careful measurements[24] on lattices of size $L=16,32,\ldots,256$ yield the dynamic critical exponent $z=0.55\pm 0.03$ (95% confidence limits). Behavior for the four-state Potts model in two dimensions, and for the Ising model in three dimensions, is only slightly less spectacular.[17,136]

Spurred by the extraordinary performance of the SW algorithm for Potts models, numerous authors[18–22] have tried in recent months to generalize the SW algorithm to non-Potts models. It is still too early to say which of these algorithms will ultimately succeed in reducing or eliminating critical slowing down for which models, but several of these algorithms offer great promise.

(1) Brower and Tamayo[23] have simulated a one-component ϕ^4 model by decomposing the spin ϕ_z into its sign $\sigma_x\equiv\text{sgn}(\phi_x)$ and its magnitude $|\phi_x|$. The signs, conditional on the magnitudes, are distributed according to a ferromagnetic Ising model (with space-dependent couplings) which can be simulated by Swendsen and Wang; and ordinary heat-bath sweeps suffice to equilibrate the magnitudes. With this algorithm, Brower and Tamayo find that the ϕ^4 model (with $\lambda\neq 0$) falls in the same dynamic universality class as the SW algorithm for the Ising model. As $\lambda\to 0$, the performance of their algorithm deteriorates; but they suggest that it could be rescued by replacing the heat-bath sweeps by the MGMC method. We have here a beautiful example of the collective-mode Monte Carlo method in action: the SW and MGMC algorithms capture the essential long-distance physics for large and small λ; and it is at least possible that a combination of the two algorithms will exhibit $z\leq z_{\text{SW}}$ (≈ 0.3 in $d=2$) *uniformly* in $0\leq\lambda\leq\infty$ (Ref. 137).

(2) Wolff[21] has devised an algorithm similar to that of Brower and Tamayo, but for *multicomponent* spin models; the initial numerical results are extremely encouraging. In particular, it appears that critical slowing down may be eliminated on *both* sides of criticality in the two-dimensional XY model. It remains unclear, however, *why* this could be the case, i.e., *how* the algorithm succeeds in creating and destroying large vortex-antivortex pairs.

(3) Niedermayer's[20] generalization of the SW algorithm may also succeed in reducing or eliminating the critical slowing down in multicomponent spin models.

(4) Brandt and collaborators[22] have described a multilevel generalization of the SW algorithm for Potts models, which may succeed in further reducing (or even eliminating) the critical slowing-down. A related algorithm is currently being investigated by Edwards, Li, and Sokal.[25]

D. Over-relaxation algorithms

A very different approach to accelerating the conventional Monte Carlo algorithms was taken by Adler,[138] and Whitmer,[39] and others;[138] the idea is to devise stochastic analogues of the successive over-relaxation (SOR) algorithm (2.28). These algorithms are not, strictly speaking, *collective-mode* algorithms, as the updates are purely local; but they do succeed in some cases in achieving strong collective effects. In the Gaussian (free-field) case, our exact analysis (Sec. VIII) shows that the dynamic critical exponent is $z=1$, provided that the over-relaxation parameter ω is tuned always to its optimal value. [This necessity of tuning ω is one of the disadvantages of SOR also for solving PDE's (Refs. 40–44).] Recent theoretical analysis by Neuberger[139] and numerical measurements by several groups[138,140] appear to confirm that $z=1$ is attainable, at least for some models.

E. Final conclusions

The collective-mode Monte Carlo method is a general philosophy, not a cut-and-dried recipe. For each model, the challenge is to find the collective modes, and then to use this physical knowledge (and some ingenuity) to invent an efficient Monte Carlo algorithm. The multigrid Monte Carlo method is one step in this direction. For

the models studied thus far, the MGMC method has produced substantial gains in efficiency: a factor of 10 in unfavorable cases, vastly more in favorable cases. Our studies are also beginning to suggest the outlines of an understanding of *why* the MGMC method works well on some models and less well on others. It seems that for models in which the dominant large-scale collective modes are spin waves, the MGMC method works excellently (critical slowing down is completely or almost completely eliminated); while for models in which the dominant collective modes have discrete elements such as spin flips (one-component ϕ^4) or vortices (two-dimensional XY), the MGMC method gains only a constant factor in efficiency (e.g., a factor of 10). On the other hand, many of the latter models are amenable to collective-mode algorithms of Swendsen-Wang type.

At the risk of closing with a standard (but true) platitude, let us note that much progress has been made in the last 2–3 years in devising new and radically more efficient Monte Carlo algorithms for quantum field theory, but that much more work—both theoretical and numerical—needs to be done in an effort to understand thoroughly the behavior of the existing algorithms and to devise new and even better ones.

ACKNOWLEDGMENTS

This work was strongly influenced by many pleasant discussions with Dan Zwanziger, to whom we gratefully record our thanks. In particular, he invented the first multigrid method for lattice gauge theories (Fig. 2 of Sec. V), which he has graciously given us permission to include here. This research was supported in part by National Science Foundation Grants Nos. PHY-8413569 and DMS-8705599. The first author (J.G.) was partially supported by the Sloan Foundation and by the Defense Advanced Research Projects Agency.

APPENDIX: HEAT-BATH ALGORITHM FOR ϕ^4 MODELS

In implementing the heat-bath phase of the MGMC algorithm for ϕ^4 models, we need a subroutine that generates a random variable X from the one-dimensional probability distribution with density proportional to

$$f(x)=\exp(-a_4x^4-a_3x^3-a_2x^2-a_1x)\ , \tag{A1}$$

where $a_4>0$ and a_1,a_2,a_3 are arbitrary. By a shift in x we can assume with loss of generality that $a_3=0$.

We use a von Neumann rejection algorithm:[141] given a function $g(x)\geq f(x)$, we generate a random variable X with density proportional to $g(x)$ and then accept it with probability $f(X)/g(X)$; we keep trying until success. The acceptance fraction is

$$A=\frac{\int_{-\infty}^{\infty}f(x)dx}{\int_{-\infty}^{\infty}g(x)dx} \tag{A2}$$

and the expected number of trials is A^{-1}.

We use a Gaussian trial distribution

$$g(x)=\exp(-\alpha x^2-\beta x-\gamma) \tag{A3}$$

($\alpha>0$) and choose α,β,γ subject to the constraint $g(x)\geq f(x)$ so as to maximize the acceptance fraction A; this means minimize

$$\int_{-\infty}^{\infty}g(x)dx\sim\alpha^{-1/2}\exp\left[\frac{\beta^2}{4\alpha}-\gamma\right]\ . \tag{A4}$$

It is not difficult to see that, for given α, the minimum area (A4) occurs when β and γ are such that the polynomial inequality

$$a_4x^4+(a_2-\alpha)x^2+(a_1-\beta)x-\gamma\geq 0 \tag{A5}$$

is satisfied with equality at two points of tangency, i.e., with *two double roots.* It follows that

$$\alpha\geq a_2,\quad \alpha>0\ , \tag{A6a}$$

$$\beta=a_1\ , \tag{A6b}$$

$$\gamma=-(\alpha-a_2)^2/4a_4\ . \tag{A6c}$$

Minimizing (A4) we then obtain a cubic equation for α,

$$\alpha^3-a_2\alpha^2-a_4\alpha-\frac{a_1^2a_4}{2}=0\ . \tag{A7}$$

It is easy to see that (A7) has precisely one real positive root. This root can be found to machine precision by a few iterations of Newton's method with the initial guess

$$\alpha^{(0)}=\frac{a_2+(a_2^2+4a_4)^{1/2}}{2}\ . \tag{A8}$$

(This is much more efficient than solving the cubic equation by the exact formula.)

The acceptance fraction of this algorithm can be computed exactly in the symmetric case $a_1=0$. In terms of the dimensionless variables $z=a_2/a_4^{1/2}$, we have $A=F(z)/G(z)$ with

$$F(z)=\begin{cases}\frac{1}{2}z^{1/2}e^{z^2/8}K_{1/4}(z^2/8) & \text{for } z>0,\\ \frac{1}{2}\Gamma(\frac{1}{4}) & \text{for } z=0,\\ \frac{1}{2}(-z)^{1/2}e^{z^2/8}[K_{1/4}(z^2/8)+\sqrt{2}\pi I_{1/4}(z^2/8)] & \text{for } z<0,\end{cases} \tag{A9}$$

$$G(z)=(\pi/2)^{1/2}(\sqrt{z^2+4}-z)^{1/2}\exp[(\sqrt{z^2+4}-z)^2/16]\ , \tag{A10}$$

where $K_{1/4}$ and $I_{1/4}$ are modified Bessel functions. In particular,

$$A = \begin{cases} 1 - \dfrac{1}{2z^2} + O\left(\dfrac{1}{z^4}\right) & \text{for } z \to +\infty \text{ (deep single-well regime)}, \\ \frac{1}{2}\pi^{-1/2} e^{-1/4} \Gamma(\frac{1}{4}) \approx 0.796 & \text{for } z = 0\ , \\ \left(\dfrac{e}{2}\right)^{1/2} (-z)^{-1}[1 + O(z^{-2})] & \text{for } z \to -\infty \text{ (deep double-well regime)}\ . \end{cases} \tag{A11}$$

Thus, the efficiency deteriorates in the deep double-well regime (as is obvious geometrically), but is tolerable for the values of z encountered in our simulation. Note that the efficiency is generally higher in the asymmetric case $a_1 \neq 0$, since one of the two wells begins to dominate.

One possible improvement in the double-well case would be to use a *double-Gaussian* trial distribution, but the details of optimizing this algorithm appear to be somewhat messy.

The special case $a_1 = a_2 = 0$ of our algorithm has been studied previously by Tadikamalla.[142] Previous Monte Carlo studies of the ϕ^4 model[143] have used Metropolis-type algorithms with various parts of the action being included in the proposal matrix. This is convenient for the multigrid Monte Carlo method, since the parameters a_1, a_2, a_3 vary from site to site on the coarse grids, making it difficult to obtain uniform control over the acceptance fraction. For this reason we have used a pure heat-bath algorithm.

---

*Electronic address: goodman@nyu.edu.BITNET.
†Electronic address: sokal@acf4.nyu.edu.BITNET.

[1] *Monte Carlo Methods in Statisitcal Physics,* 2nd ed., edited by K. Binder (Springer, Berlin, 1986); *Applications of the Monte Carlo Method in Statistical Physics,* 2nd ed., edited by K. Binder (Springer, Berlin, 1987); D. Stauffer, J. Appl. Phys. **53**, 7980 (1982); K. Binder, J. Comput. Phys. **59**, 1 (1985).

[2] M. Creutz, L. Jacobs, and C. Rebbi, Phys. Rep. **95**, 201 (1983); *Lattice Gauge Theories and Monte Carlo Simulations,* edited by C. Rebbi (World Scientific, Singapore, 1983).

[3] Each block of data of length $\approx 2\tau$ can be considered to contribute one "statistically independent" data point. Therefore, the "effective sample size" from a run of N sweeps is only $\approx N/2\tau$, and the statistical error bar is of order $(\tau/N)^{1/2}$. For a more detailed treatment, see Sec. VI. See also K. Binder, in *Monte Carlo Methods in Statistical Physics* (Ref. 1), Secs. 1.2.3 and 1.2.4; A. Berretti and A. D. Sokal, J. Stat. Phys. **40**, 483 (1985), Sec. 4.1; N. Madras and A. D. Sokal, J. Stat. Phys. **50**, 109 (1988), Sec. 2.2 and Appendix C.

[4] A brief description was given in J. Goodman and A. D. Sokal, Phys. Rev. Lett. **56**, 1015 (1986). Several other authors have also made proposals for multigrid Monte Carlo methods; we discuss these proposals in Sec. X A.

[5] R. G. Edwards, J. Goodman, and A. D. Sokal (unpublished).

[6] R. G. Edwards, J. Goodman, and A. D. Sokal (in preparation).

[7] R. G. Edwards, J. Goodman, D. Ritchie, A. D. Sokal, and D. Zwanziger (in preparation).

[8] The multigrid method is discussed, for example, in A. Brandt, Math. Comput. **31**, 333 (1977); *Multigrid Methods,* edited by W. Hackbusch and U. Trottenberg (Lecture Notes in Mathematics No. 960) (Springer, Berlin, 1982); W. Hackbusch, *Multigrid Methods and Applications* (Springer, Berlin, 1985); W. L. Briggs, *A Multigrid Tutorial* (SIAM, Philadelphia, 1987). We give a pedagogical introduction to multigrid methods in Sec. II of this paper.

[9] The resemblance between the multigrid method and the renormalization group has often been noted: see, e.g., O. A. McBryan, Physica **124A**, 481 (1984), p. 488.

[10] The idea of "collective mode Monte Carlo" was proposed by M. Kalos, in proceedings of the Brookhaven Conference on Monte Carlo Methods and Future Computer Architecture, 1983 (unpublished). Very similar ideas were proposed by G. Parisi, in *Progress in Gauge Field Theory,* 1983 Cargèse Lectures, edited by G. 't Hooft *et al.* (Plenum, New York, 1984).

[11] G. Parisi, in *Progress in Gauge Field Theory* (Ref. 10); G. G. Batrouni, G. R. Katz, A. S. Kronfeld, G. P. Lepage, B. Svetitsky, and K. G. Wilson, Phys. Rev. D **32**, 2736 (1985); C. Davies, G. Batrouni, G. Katz, A. Kronfeld, P. Lepage, P. Rossi, B. Svetitsky, and K. Wilson, J. Stat. Phys. **43**, 1073 (1986); E. Dagotto and J. B. Kogut, Nucl. Phys. **B290** [FS20], 451 (1987).

[12] J. B. Kogut, Nucl. Phys. **B275** [FS17], 1 (1986); S. Duane, R. Kenway, B. J. Pendleton, and D. Roweth, Phys. Lett. B **176**, 143 (1986); E. Dagotto and J. B. Kogut, Phys. Rev. Lett. **58**, 299 (1987); Nucl. Phys. **B290** [FS20], 451 (1987).

[13] Indeed, for the Gaussian model this random-walk picture can be made rigorous; see J. Goodman and N. Madras, Courant Institute report, 1987 (unpublished), combined with Sec. VII of this paper.

[14] P. C. Hohenberg and B. I. Halperin, Rev. Mod. Phys. **49**, 435 (1977); J. Zinn-Justin, Nucl. Phys. **B275** [FS17], 135 (1986).

[15] For infrared-superrenormalizable models (e.g., ϕ^4 in dimension $d > 4$), we have $z = 2$ exactly. For infrared-asymptotically-free theories (e.g., ϕ^4 in dimension $d = 4$), we have $z = 2$ modulo a logarithm. For Ising and ϕ^4 models in dimensions $d = 2$ and 3, numerical estimates of z can be found in G. F. Mazenko and O. T. Valls, Phys. Rev. B **24**, 1419 (1981); C. Kalle, J. Phys. A **17**, L801 (1984); J. K. Williams, *ibid.* **18**, 49 (1985); R. B. Pearson, J. L. Richardson, and D. Toussaint, Phys. Rev. B **31**, 4472 (1985); S. Wansleben and D. P. Landau, J. Appl. Phys. **61**, 3968 (1987).

[16] Clearly one must taken $L \gtrsim \xi$ in order to avoid severe finite-size effects. Typically one approaches the critical point with $L \approx c\xi$, where $c \approx 2$–4.

[17] R. H. Swendsen and J.-S. Wang, Phys. Rev. Lett. **58**, 86 (1987).

[18] U. Wolff, Phys. Rev. Lett. **60**, 1461 (1988); Nucl. Phys. **B300** [FS22], 501 (1988).

[19] R. G. Edwards and A. D. Sokal, Phys. Rev. D **38**, 2009 (1988).

[20] F. Niedermayer, Phys. Rev. Lett. **61**, 2026 (1988).

[21] U. Wolff, Phys. Rev. Lett. **62**, 361 (1989); U. Wolff, DESY Report No. 88-176, 1988 (unpublished); Phys. Lett. B **222**, 473 (1989); DESY Report No. 89-021, 1989 (unpublished).

[22] A. Brandt, in *Proceedings of the Third Copper Mountain*

Conference on Multigrid Methods, edited by S. McCormick (Dekker, New York, 1988); D. Kandel, E. Domany, D. Ron, A. Brandt, and E. Loh, Jr., Phys. Rev. Lett. **60**, 1591 (1988).

[23]R. C. Brower and P. Tamayo, Phys. Rev. Lett. **62**, 1087 (1989).

[24]R. G. Edwards and A. D. Sokal (in preparation).

[25]R. G. Edwards, X.-J. Li, and A. D. Sokal (in preparation).

[26]R. P. Fedorenko, Zh. Vychisl. Mat. Fiz. **4**, 559 (1964) [USSR Comput. Math. Phys. **4**, 227 (1964)]. Three years earlier, Fedorenko had taken the first step by proposing a "two-grid" algorithm: see Zh. Vychisl. Mat. Mat. Fiz. **1**, 92 (1961) [USSR Comput. Math. Math. Phys. **1**, 1092 (1961)]. This latter paper is very clear and is still worth reading today.

[27]L. P. Kadanoff, Physics **2**, 263 (1966).

[28]K. G. Wilson, Phys. Rev. B **4**, 3174 (1971); **4**, 3184 (1971).

[29]A. Brandt, in *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics,* Paris, France, 1972, edited by H. Cabannes and R. Temam (Lecture Notes in Physics No. 18) (Springer, Berlin, 1973); R. A. Nicolaides, J. Comput. Phys. **19**, 418 (1975); Math. Comput. **31**, 892 (1977); A. Brandt, *ibid.* **31**, 333 (1977); W. Hackbusch, in *Numerical Treatment of Differential Equations,* Oberwolfach, 1976, edited by R. Bulirsch, R. D. Griegorieff, and J. Schröder (Lecture Notes in Mathematics No. 631) (Springer, Berlin, 1978).

[30]W. Hackbusch, *Multi-Grid Methods and Applications* (Springer, Berlin, 1985).

[31]*Multigrid Methods,* edited by W. Hackbusch and U. Trottenberg (Lecture Notes in Mathematics No. 960) (Springer, Berlin, 1982).

[32]*Proceedings of the First Copper Mountain Conference on Multigrid Methods,* edited by S. McCormick and U. Trottenberg [Appl. Math. Comput. **13**, 213 (1983)].

[33]*Multigrid Methods for Integral and Differential Equations,* edited by D. J. Paddon and H. Holstein (Clarendon, Oxford, 1985).

[34]*Proceedings of the Second Copper Mountain Conference on Multigrid Methods,* edited by S. McCormick [Appl. Math. Comput. **19**, 1 (1986)].

[35]*Multigrid Methods II,* edited by W. Hackbusch and U. Trottenberg (Lecture Notes in Mathematics No. 1228) (Springer, Berlin, 1986).

[36]*Multigrid Methods,* edited by S. F. McCormick (SIAM, Philadelphia, 1987).

[37]*Proceedings of the Third Copper Mountain Conference on Multigrid Methods* (Ref. 22).

[38]S. L. Adler, Phys. Rev. D **23**, 2901 (1981).

[39]C. Whitmer, Phys. Rev. D **29**, 306 (1984).

[40]R. S. Varga, *Matrix Iterative Analysis* (Prentice-Hall, Englewood Cliffs, NJ, 1962).

[41]H. R. Schwarz, H. Rutishauer, and E. Stiefel, *Numerical Analysis of Symmetric Matrices* (Prentice-Hall, Englewood Cliffs, NJ, 1973).

[42]E. L. Wachspress, *Iterative Solution of Elliptic Systems and Applications to the Neutron Diffusion Equations of Reactor Physics* (Prentice-Hall, Englewood Cliffs, NJ, 1966).

[43]D. M. Young and R. T. Gregory, *A Survey of Numerical Mathematics* (Addison-Wesley, Reading, MA, 1973), Vol. II, Chap. 11.

[44]D. M. Young, *Iterative Solution of Large Linear Systems* (Academic, New York, 1971).

[45]W. L. Briggs, *A Multigrid Tutorial* (Ref. 8).

[46]A. Brandt, Math. Comput. **31**, 333 (1977); in *Multigrid Methods* (Ref. 31); in *Multigrid Methods for Integral and Differential Equations* (Ref. 33).

[47]J. Goodman and A. D. Sokal (in preparation). For a sketch of the convergence proofs for MG and Gaussian MGMC methods with piecewise-constant interpolation, see J. Goodman, in *Proceedings of the Third Copper Mountain Conference on Multigrid Methods* (Ref. 37).

[48]In most applications, (2.1) would arise as the discretization of a continuum Laplace equation using some small lattice spacing a. In that case the total error would be the sum of two contributions: the error in solving (2.1), plus the discretization error [i.e., the accuracy with which the solution of (2.1) approximates the solution of the original continuum equation]. The tradeoff between these two contributions affects both the choice of algorithm and the choice of the lattice spacing a. In this paper, however, we take a different point of view: we consider the lattice equation (2.1) as our starting point, without inquiring where it came from. This point of view is useful in quantum field theory, where ultraviolet divergences spoil the naive correspondence between continuum and lattice quantities.

[49]J. M. Ortega, *Numerical Analysis: A Second Course* (Academic, New York, 1972), pp. 118–119, Theorem 7.1.1. In fact, it is not hard to show that $\rho(M)$ is the precise worst-case asymptotic rate of convergence (and not just an upper bound), in the sense that

$$\sup_{\phi^{(0)}} \limsup_{n\to\infty} \|\phi^{(n)}-\phi\|^{1/n} = \rho(M) \ .$$

See Ortega, Theorem 7.2.2, p. 126.

[50]This follows from Varga, *Matrix Iterative Analysis* (Ref. 40), p. 73, Theorem 3.4, since the matrix $-\Delta$ with Dirichlet boundary conditions is irreducibly diagonally dominant. Alternatively, it follows from p. 70, Theorem 3.3, and p. 78, Corollary 1, because $-\Delta$ is positive definite and has nonpositive off-diagonal entries.

[51]J. Goodman and N. Madras, Courant Institute report, 1987 (unpublished).

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

[53]We define the algorithm recursively because that is by far the simplest and clearest approach. Of course, the algorithm can be implemented in any computer programming language, including those (like FORTRAN) that forbid recursive subroutine calls, although the control structure may be a bit more complicated. See Hackbusch, *Multi-Grid Methods and Applications* (Ref. 30), Sec. 4.1 for FORTRAN and ALGOL implementations.

[54]For further discussion of coarsening strategies, see Hackbusch, *Multi-Grid Methods and Applications* (Ref. 30), Secs. 3.4.1, 3.8.2, 3.8.4, and 10.5. A totally different approach to coarsening is taken by the algebraic multigrid (AMG) method: see J. W. Ruge and K. Stüben, in *Multgrid Methods* (Ref. 36), Chap. 4.

[55]The trivial restriction is very reminiscent of the use of *decimation* as a renormalization-group transformation, which is known to suffer from severe difficulties [L. P. Kadanoff and A. Houghton, Phys. Rev. B **11**, 377 (1975)]. Intuitively the trouble is that no coarse graining (averaging) is being performed, so that low-frequency components are not in any way being favored over high-frequency ones. In the multigrid context, however, this is not necessarily a devastating criticism (at least for equations with constant or slowly varying coefficients), since the high-frequency components have already been suppressed by the presmoothing. The viability of trivial restriction thus depends on a delicate interplay be-

tween the smoothing and the coarse-grid correction. For an example of how trivial restriction can lead to disaster, see Hackbusch, *Mult-Grid Methods and Applications* (Ref. 30), p. 64; for an example of how it can work well, see D. Braess, Numer. Math. **37**, 387 (1981); Math. Comput. **42**, 505 (1984).

[56]See Hackbusch, *Multi-Grid Methods and Applications* (Ref. 30), p. 69 and p. 79, Exercises 3.9.4–6.

[57](2.27) is called the Gauss-Seidel iteration because "Gauss did not use a cyclic order of relaxation, and . . . Seidel specifically recommended against using it" [Schwarz, Rutishauer, and Steifel, *Numerical Analysis of Symmetric Matrices* (Ref. 41), p. 44ff]. See also A. M. Ostrowski, Rendiconti Mat. Appl. **13**, 140 (1954), p. 141ff [A Ostrowski, *Collected Mathematical Papers* (Birkhaüser, Basel, 1983), Vol. 1, p. 169ff].

[58]Varga, *Matrix Iterative Analysis* (Ref. 40), Sec. 3.4.

[59]Varga, *Matrix Iterative Analysis* (Ref. 40), Secs. 4.2, 4.4, and 4.5. See also Goodman-Madras (Ref. 51), for a random-walk interpretation of this result.

[60]K. Stüben and U. Trottenberg, in *Multigrid Methods* (Ref. 31), p. 128.

[61]See Hackbusch, *Multi-Grid Methods and Application* (Ref. 30), Sec. 3.3 and Chaps. 10 and 11.

[62]Hackbusch, *Multi-Grid Methods and Applications* (Ref. 30), Sec. 4.3.

[63]For a detailed exposition of multigrid convergence proofs, see Hackbusch, *Multi-Grid Methods and Applications* (Ref. 30), Chaps. 6–8, 10, 11, and J. Mandel, S. McCormick, and R. Bank, in *Multigrid Methods* (Ref. 36), Chap. 5, and references cited therein. The additional work needed to handle the piecewise-constant interpolation can be found in Goodman and Sokal (Ref. 47).

[64]Indeed, in most cases the space V has a natural *physical* interpretation as the dual space of U. For example, if the elements of U are displacements, electric potentials, or magnetizations, then the elements of V are usually forces, charge densities, or magnetic fields, respectively; and the natural pairing $\langle f,\phi\rangle$ of $\phi\in U$ with $f\in V=U^*$ yields an *energy*. Moreover, in most cases the operator A is symmetric positive definite.

[65]J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables* (Academic, New York/London, 1970), pp. 512–520.

[66]S. F. McCormick and J. Ruge, Math. Comput. **41**, 43 (1983).

[67]Only about *half* of the energy of a slowly varying function can be removed by a piecewise-constant update. (This is easily seen in one dimension, where half of the nearest-neighbor bonds are unaffected by a piecewise-constant update, while the energy on the other half can be eliminated entirely.) It turns out (Ref. 47) that this degree of approximation is good enough for W cycle convergence, but not for V cycle convergence.

[68]Hackbusch, *Multi-Grid Methods and Applications* (Ref. 30), Chap. 9.

[69]In practice we eliminate the ϕ^3 term by a shift of the field (see the Appendix), but this is purely a question of computational convenience, not of principle.

[70]Ortega and Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables* (Ref. 65).

[71]Ortega and Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables* (Ref. 65), pp. 219, 244–245, 249, 513–517.

[72]Ortega and Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables* (Ref. 65), pp. 299–310.

[73]See D. Zwanziger, in *Fundamental Problems of Gauge Field Theory,* proceedings of the International School, Erice, Italy, 1985, edited by G. Velo and A. S. Wightman (NATO ASI Series B, Vol. 141) (Plenum, New York, 1987); C. T. H. Davies, G. G. Batrouni, G. R. Katz, A. S. Kronfeld, G. P. Lepage, K. G. Wilson, P. Rossi, and B. Svetitsky, Phys. Rev. D **37**, 1581 (1988).

[74]The XY model with frustration has also some intrinsic interest: see M. Y. Choi and S. Doniach, Phys. Rev. B **31**, 4516 (1985); M. Y. Choi and D. Stroud, *ibid.* **32**, 5773 (1985); **32**, 7532 (1985); J. M. Thijssen and H. J. F. Knops, *ibid.* **37**, 7738 (1988), and references cited therein.

[75]We remark that the piecewise-linear interpolation (2.5) is not so easily extended to the XY model: how does one define $\frac{1}{2}$ or $\frac{1}{4}$ of an angular variable? The answer is that the configuation space at each site on grid Ω_l should be the group of real numbers modulo $4^{M-l}\times 2\pi$ (rather than 2π); then the fractional maps $\frac{1}{2}$ and $\frac{1}{4}$ *are* well-defined from a circle of width K to a circle of width $K/4$. However, the coarse-grid Hamiltonians become quite complicated.

[76]See, e.g., A. M. Polyakov, *Gauge Fields and Strings* (Harwood Academic, Chur, Switzerland, 1987).

[77]We remark that since most of the work in a multigrid iteration takes place on the finest grid [cf. (2.29) and (2.30)], it is usually more efficient *in practice* to treat the finest grid separately, in order to taken advantage of the simpler form of its Hamiltonian [(4.10) versus (4.5)]. In that case, the fine-grid theory would live on the manifold M, while all the coarse-grid theories would live on the group G.

[78]Let us emphasize that there is nothing in principle *invalid* about a multigrid algorithm which makes piecewise-constant (rather than piecewise-*covariant*-constant) moves. Rather, the point is a multigrid algorithm is *efficient* only if the coarse-grid-correction moves reflect the important collective modes of the system. It seems clear to us that the correct collective modes of a gauge theory should be defined in a gauge-invariant way.

[79]This has also been pointed out, in the context of Fourier acceleration, S. Duane, R. Kenway, B. J. Pendleton, and D. Roweth, Phys. Lett. B **176**, 143 (1986).

[80]The V's may themselves be obtained by parallel transport of other matrices, e.g., $V_{AB}=U_{12}^{-1}VU_{12}$ as described above. But this fact neither complicates nor simplifies the form of the expression.

[81]We thank Andrea Pelissetto for this observation.

[82]A *probability kernel* is an object $P=P(\phi, A)$ whose first argument is a point $\phi\in S$ and whose second argument is a (measurable) set $A\subset S$, such that (a) $P(\phi, A)$ is a measurable function of ϕ for each fixed A; and (b) $P(\phi, A)$ is a probability measure in A, for each fixed ϕ. Powers of kernel are defined analogously to powers of matrices. Indeed, if the state space S is countable, there is a one-to-one correspondence between kernels and matrices. The Markov chain with transition probability kernel P is defined as follows: if ϕ is the state of the system at time t, then the state of the system at time $t+1$ is chosen randomly according to the probability $P(\phi,\cdot)$.

[83]We avoid the term "ergodicity" because of its multiple and conflicting meanings. In the physics literature, and in the mathematics literature on Markov chains with *finite* state space, "ergodic" is typically used as a synonym for "irreducible" [J. G. Kemeny and J. L. Snell, *Finite Markov Chains* (Springer, New York, 1976), Sec. 2.4; M. Iosifescu, *Finite Markov Processes and Their Applications* (Wiley, Chichester, 1980), Chap. 4]. However, in the mathematics literature on Markov chains with *general* state space "ergodic" is used as a

synonym for "irreducible, aperiodic, and positive Harris recurrent" [E. Nummelin, *General Irreducible Markov Chains and Non-Negative Operators* (Cambridge University Press, Cambridge, England, 1984), p. 114; D. Revuz, *Markov Chains* (North-Holland, Amsterdam, 1975), p. 169].

[84]For Markov chains with *finite* state space, see Kemeny and Snell, *Finite Markov Chains* (Ref. 83) and Iosifescu, *Finite Markov Processes and Their Applications* (Ref. 83); for Markov chains with *countable* state space, see K. L. Chung, *Markov Chains with Stationary Transition Probabilities,* 2nd ed. (Springer, New York, 1967); for Markov chains with *general* state space, see Nummelin, *General Irreducible Markov Chains and Non-Negative Operators* (Ref. 83), and Revuz, *Markov Chains* (Ref. 83).

[85]In the statistics literature, this is called the *autocovariance function.*

[86]Z. Šidak, Czech. Math. J. **14**, 438 (1964); **17**, 148 (1967). Some necessary lemmas can be found in Nummelin, *General Irreducible Markov Chains and Non-Negative Operators* (Ref. 83), Chaps. 2 and 3.

[87]A. D. Sokal and L. E. Thomas, J. Stat. Phys. **54**, 797 (1989).

[88]In the mathematical literature, a Markov chain satisfying the detailed-balance condition is called *reversible.* For the physical significance of this term, see Kemeny and Snell [*Finite Markov Chains* (Ref. 83), Sec. 5.3] or Iosifescu [*Finite Markov Processes and Their Applications* (Ref. 83), Sec. 4.5]. Unfortunately, the term "reversible" is also used in the physics literature on Monte Carlo methods with a *different* meaning [e.g., S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, Phys. Lett. B **195**, 216 (1987)].

[89]The existence of a decomposition (7.3) with the above properties is a nontrivial mathematical fact (needing certain regularity hypotheses): it is called variously the *quotient measure theorem,* the theorem on *disintegration of measures,* or the existence theorem for *proper regular conditional probabilities.* See K. R. Parthasarathy, *Probability Measures on Metric Spaces* (Academic, New York, 1967), Sec. V.8; D. Blackwell and L. E. Dubins, Ann. Probab. **3**, 741 (1975); D. Ramachandran, *ibid* **7**, 433 (1979); A. D. Sokal, Z. Wahrscheinlichkeitstheorie, Verw. Geb. **56**, 537 (1981).

[90]L. Nachbin, *The Haar Integral* (Van Nostrand, Princeton, 1965).

[91]Some of this material appears also in S. L. Adler, Phys. Rev. Lett. **60**, 1243 (1988).

[92]The heat-bath algorithm with *random* choice of sites is not immediately covered by the analysis given below, rather, its transition probability kernel P is a *convex combination* of those treated here. However, it could possibly by analyzed by a suitable extension of our methods.

[93]We remark that by Stein's theorem [Ortega, *Numerical Analysis: A Second Course* (Ref. 49), p. 122, Theorem 7.1.8], (8.4b) can be satisfied only if $\rho(M) \le 1$. *Proof:* Let λ be any eigenvalue of M^T, and ψ the corresponding eigenvector. Then take the inner product of (8.4b) with ψ on both sides; we get $1-|\lambda|^2 \ge 0$, with strict inequality if QCQ^T is nonsingular.

[94]If C is nonsingular, a matrix Q satisfying (8.4b) always exists. But it is in general highly nonunique.

[95]We remark that this verification never uses the fact that D is diagonal or that L is lower triangular. It is sufficient to have $A = D + L + L^T$ with D symmetric positive definite and $D+L$ nonsingular. For the method to be practical, it is important that $D^{1/2}$ and $(D+L)^{-1}$ be "easy" to compute when applied to a vector.

[96]With the choice $Q = \delta^{1/2}[I - (\delta/4)C]^{1/2}$ Eq. (8.4b) is satisfied exactly. This choice is possible only for $\delta \le 4\|C\|^{-1}$.

[97]This is also remarked by Adler (Ref. 91).

[98]E. Nelson, in *Constructive Quantum Field Theory,* edited by G. Velo and A. Wightman (Lecture Notes in Physics No. 25) (Springer, Berlin, 1973); B. Simon, *The P $(\phi)_2$ Euclidean (Quantum) Field Theory* (Princeton University Press, Princeton, NJ, 1974), Chap. I.

[99]We remark that (8.23) is essentially a multidimensional Mehler's formula. See, e.g., Simon, *The P $(\phi)_2$ Euclidean (Quantum) Field Theory* (Ref. 98), p. 29.

[100]Varga, *Matrix Iterative Analysis* (Ref. 40), Chap. 4 and Sec. 6.5. For a simpler derivation (using Fourier methods), see, e.g., H. Neuberger, Phys. Rev. Lett. **59**, 1877 (1987); R. J. LeVeque and L. N. Trefethen, IMA J. Numerical Anal. **8**, 273 (1988).

[101]Recall Eq. (2.30). We also confirmed this empirically by making timing measurements on our program.

[102]H. J. Brascamp and E. H. Lieb, in *Functional Integration and its Applications,* edited by A. M. Arthurs (Clarendon, Oxford, 1975); H. J. Brascamp and E. H. Lieb, J. Funct. Anal. **22**, 366 (1976).

[103]Let $\mathcal{A}$ be the random variable in brackets in (9.6a), i.e.,

$$\begin{aligned}\mathcal{A} &\equiv \frac{1}{b^d}\sum_{x\in B}(\phi_x - \bar{\phi}_x)^2 \\ &= \frac{1}{b^d}\sum_{x\in B}\phi_x^2 - \frac{1}{b^{2d}}\left[\sum_{x\in B}\phi_x\right]^2 \\ &= \mathcal{A}_0 - \frac{1}{b^{2d}}\left[\sum_{x\in B}\phi_x\right]^2 .\end{aligned}$$

Then the mean value of $\mathcal{A}$ is

$$\langle \mathcal{A} \rangle = \langle \phi_x^2 \rangle - b^{-d}\chi_B ,$$

where

$$\chi_B \equiv \frac{1}{b^d}\sum_{x,y\in B}\langle \phi_x \phi_y \rangle$$

is $\ll b^d$ provided that we are in the single-phase region (i.e., above or at the critical temperature). Indeed, χ_B is a finite-volume approximation to the susceptibility, so that we can expect $\chi_B \sim b^{2-\eta}$ for b comparable to the correlation length ξ. Therefore $\langle \mathcal{A} \rangle \approx \langle \phi_\lambda^2 \rangle$ for large blocks. On the other hand, in the single-phase region, the variance of $\mathcal{A}$ is presumably roughly equal to the variance of $\mathcal{A}_0$, which is

$$\mathrm{var}(\mathcal{A}_0) = \frac{1}{b^{2d}}\sum_{x,y\in B}\langle \phi_x^2; \phi_y^2 \rangle \equiv \frac{1}{b^d}\kappa_B .$$

Here κ_B is a finite-volume approximation to what is essentially the specific heat, so we expect that $\kappa_B \sim b^{\alpha/\nu}$ for b comparable to the correlation length ξ. This is a very weak divergence. In particular, the standard deviation of $\mathcal{A}$ is vastly smaller than its mean (when $b \gg 1$), justifying the claim that $\mathcal{A}$ is with very high probability close to its mean. *Final remark:* Using the Griffiths II and Lebowitz inequalities [see, e.g., G. S. Sylvester, J. Stat. Phys. **15**, 327 (1976)], it is possible to analyze $\mathrm{var}(\mathcal{A})$ directly, proving the upper bound $\mathrm{var}(\mathcal{A}) \le b^{-d}\kappa_B + 2b^{-2s}\chi_B^2$. The final term here is presumably spurious, but even this crude estimate justifies rigorously the claim that (standard deviation of $\mathcal{A}$) $\ll \langle \mathcal{A} \rangle$.

[104]G. S. Sylvester, J. Stat. Phys. **15**, 327 (1976).

[105]*Claim:* In the measure $d\nu(\phi) \sim \exp(-\lambda\phi^4 - B\phi^2)$ with $\lambda > 0$, we have $\langle \phi^2 \rangle \ge -B/2\lambda$. *Proof:* The only nontrivial case is

$B<0$; and by a rescaling of ϕ, it suffices to consider the measure $d\nu(\phi)\sim\exp[-\lambda(\phi^2-1)^2]$ and prove that $\langle\phi^2\rangle\geq 1$. Now, for large λ, we have $\langle\phi^2\rangle=1+(8\lambda)^{-1}+O(\lambda^{-2})$. On the other hand, we know that

$$\frac{d}{d\lambda}\langle\phi^2\rangle=-\langle\phi^2;(\phi^2-1)^2\rangle$$
$$=-\langle\phi^6\rangle+2\langle\phi^4\rangle+\langle\phi^4\rangle\langle\phi^2\rangle-2\langle\phi^2\rangle^2 .$$

Using the Schwinger-Dyson identity

$$\langle F'(\phi)\rangle=4\lambda\langle F(\phi)(\phi^3-\phi)\rangle$$

with $F(\phi)=\phi^3$ and ϕ, we eliminate $\langle\phi^6\rangle$ and $\langle\phi^4\rangle$ to obtain

$$\frac{d}{d\lambda}\langle\phi^2\rangle=-\langle\phi^2\rangle(\langle\phi^2\rangle-1)-\frac{1}{4\lambda}(2\langle\phi^2\rangle-1) .$$

Integrating this differential equation from large λ downwards, we conclude that $(d/d\lambda)\langle\phi^2\rangle\leq 0$ and $\langle\phi^2\rangle\geq 1$ for all $\lambda>0$.

[106]This reasoning is a bit weak: for if (9.11) were the *true* behavior of $\mathrm{var}(t)$—and not merely an *upper bound* for it—then quite possibly the critical slowing down could be overcome by using a "super W cycle" (i.e., γ sufficiently large). After all, in the Gaussian theory we know that the "correct" amplitude for large-scale collective modes is $\mathrm{var}(t)\sim 1/b^{d-2}$, and yet, the smaller amplitude given by (9.11), namely $\mathrm{var}(t)\sim 1/b^{d-1}$ (which arises from the fact that square waves are "not smooth enough"), is sufficient for overcoming the critical slowing-down *provided* that at least a W cycle is used (i.e., $\gamma\geq 2$). Therefore, one might expect analogously that $\mathrm{var}(t)\sim 1/b^d$ could be compensated by making γ yet larger (i.e., by making *more* coarse-grid updates to compensate for the lesser effectiveness of each of them). However, we suspect that this is *not* in fact the case, and that the behavior of the MGMC method for ϕ^4 theories is even *worse* than (9.11) would suggest; more precisely, we suspect that successive updates are anticorrelated in such a way that many small moves do *not* have the same net effect as one big move. This is certainly what happens at large λ; and it seems plausible to guess that it happens for *all* λ, albeit at a length scale b which grows to infinity as $\lambda\to 0$. But we do not have any really good argument to support this guess. Also, it should be noted that our assessment of powers of b (e.g., b^{d-2} vs b^{d-1}) is based on our experience with the Gaussian theory; we do not really know how big an update the *interacting* (non-Gaussian) theory "wants." So the quantitative predictions of our argument should probably not be taken too seriously. But we expect that the qualitative conclusions are still correct.

[107]Note that the critical point of a ϕ^4 model is *always* in the double well (nonvex) regime $A<0$: the ϕ^4 coupling induces a *strictly positive* mass renormalization that needs to be compensated by a negative bare mass squared. [For a proof at small λ, see J. Bricmont, J.-R. Fontaine, and E. Speer, Commun. Math. Phys. **86**, 337 (1982) or D. C. Brydges, J. Fröhlich, and A. D. Sokal, *ibid.* **91**, 141 (1983). The result for arbitrary $\lambda>0$ then follows by Griffiths' second inequality.] In fact, general arguments based on the Brascamp-Lieb inequality (Ref. 102) show that any model with a "uniformly strictly convex" Hamiltonian, i.e., $H=H_1+\epsilon\sum_x\phi_x^2$ with H_1 convex and $\epsilon>0$, *must* be "massive" in the sense that the connected two-point function in momentum space is uniformly bounded: $0\leq\tilde{G}(p)\leq 1/2\epsilon<\infty$ for all p.

[108]H. J. Brascamp, E. H. Lieb, and J. L. Lebowitz, Bull. Internat. Stat. Inst. **46**, Book 1, 393 (1975); J.-R. Fontaine, Commun. Math. Phys. **87**, 385 (1982); K. Gawedzki and A. Kupiainen, Ann. Phys. (N.Y.) **147**, 198 (1983); and references cited therein.

[109]V. L. Berezinskii, Zh. Eksp. Teor. Fiz. **59**, 907 (1970) [Sov. Phys. JETP **32**, 493 (1971)]; **61**, 1144 (1972) [**34**, 610 (1972)]; J. M. Kosterlitz and D. J. Thouless, J. Phys. C **6**, 1181 (1973); J. M. Kosterlitz, *ibid.* **7**, 1046 (1974); J. V. José, L. P. Kadanoff, S. Kirkpatrick, and D. R. Nelson, Phys. Rev. B **16**, 1217 (1977); R. Savit, *ibid.* **17**, 1340 (1978); Rev. Mod. Phys. **52**, 453 (1980); J. Fröhlich and T. Spencer, Commun. Math. Phys. **81**, 527 (1981).

[110]See, e.g., Polyakov, *Gauge Fields and Strings* (Ref. 76), Chap. 2.

[111]The estimation of the autocorrelation time $\tau_{\mathrm{int},F}$ (together with valid statisical error bars) is discussed, for example, in N. Madras and A. D. Sokal, J. Stat. Phys. **50**, 109 (1988), Appendix C. The exponential autocorrelation time $\tau_{\mathrm{exp},F}$ is of some *theoretical* interest, but it does not play a primary role in determining the efficiency of Monte Carlo algorithm. Moreover, $\tau_{\mathrm{exp},F}$ is *extremely* difficult to measure accurately, it is defined in terms of the $t\to\infty$ tail of the autocorrelation function $\rho_{FF}(t)$, but that is exactly where the "signal-to-noise ratio" goes to zero exponentially rapidly [see, e.g., Madras and Sokal, Eq. (C9)]. Therefore, good measurements of $\tau_{\mathrm{exp},F}$ might need runs of length $10^6\tau$ or more, while good measurements of $\tau_{\mathrm{int},F}$ can usually be obtained with runs of length $10^3\tau$–$10^4\tau$.

[112]Parisi, in *Progress in Gauge Field Theory* (Ref. 10).

[113]G. Gallavotti, Mem. Accad. Lincei **15**, 23 (1978); Ann. Mat. Pura Appl. **120**, 1 (1979); G. Benfatto, M. Cassandro, G. Gallavotti, F. Nicolò, E. Olivieri, E. Presutti, and E. Scacciatelli, Commun. Math. Phys. **59**, 143 (1978); **71**, 95 (1980); G. Benfatto, G. Gallavotti, and F. Nicolò, J. Funct. Anal. **36**, 343 (1980).

[114]K. Gawedzki and A. Kupiainen, Commun. Math. Phys. **77**, 31 (1980); Ann. Phys. (N.Y.) **147**, 198 (1983); Commun. Math. Phys. **99**, 197 (1985), and many other papers. For an excellent review, see K. Gawedzki and A. Kupiainen, in *Critical Phenomena, Random Systems, Gauge Theories,* proceedings of the Les Houches Summer School, Les Houches, France, 1984, edited by K. Osterwalder and R. Stora (Les Houches Summer School Proceedings Vol. 43) (North-Holland, Amsterdam, 1986), Part I.

[115]G. Mack, in *Nonperturbative Quantum Field Theory,* 1987 Cargèse Lectures, edited by G. 't Hooft *et al.* (Plenum, New York, 1988).

[116]This difference between piecewise-constant and higher-order interpolations may be what Mack is getting at ". . . it is wise to follow the path outlined by successful rigorous analytical work, as we did, and to reduce coupling between layers as much as possible. The simple proposal [of Goodman and Sokal] to change the field ϕ by amounts that are constant on cubes $x\in\Lambda_j$ fails to do so. It leads to couplings between layers j and $k>j$ which are too big by a factor L^{k-j}. Therefore this is not a good choice . . ." (Ref. 115, Sec. II.1) The conclusion of this argument is correct for a V cycle, but *not* for a W cycle.

[117]"The fast approach to correct conditional probability distributions for the fields $\phi^{(j)}$ on individual layers comes from the fast decay of correlation functions $\langle\phi_x^{(j)}\phi_y^{(j)}\rangle$. [But this] is not sufficient to guarantee that the whole system approaches equilibrium fast, because $\phi_x^{(j)}$ may be strongly correlated with many variables $\phi_y^{(k)}$ with $k>j$ . . . . Suppose that the coupling between layers is so strong that $\phi^{(j)}$ is essentially deter-

mined by the $\phi^{(k)}$'s with $k > j$. Then correct conditional probability distributions for $\phi^{(j)}$ are trivially attained, but from this one learns nothing about the approach of the whole system to equilibrium. . . . For example, deep in the two-phase region the fluctuation field $\zeta^{(0)}$ (which is given by $\phi^{(k)}$'s with $k > 0$) 'knows' the state of the system. It determines the (sign of) magnetization $\Phi^{(0)} = \phi^{(0)}$. If there were no coupling between $\zeta^{(0)}$ and $\Phi^{(0)}$, updating $\Phi^{(0)}$ would give a tunneling probability of 50%, because $\mathcal{H}^0(\Phi^{(0)})$ is symmetric. The observed tunneling rate is much lower." (Ref. 115, Sec. II.1.)

[118]G. Mack and S. Meyer (in preparation).

[119]H. Meyer-Ortmanns, Z. Phys. C **27**, 553 (1985). We became aware of this paper only quite recently, thanks to the citation in Ref. 115.

[120]A. Brandt, D. Ron, and D. J. Amit, in *Multigrid Methods II* (Ref. 35). See Sec. 7.1.

[121]"This includes, for every coarse-grid configuration u^c, the energy function $E^c(u^c) = E^f(I_c^f u^c)$. Since this coarse-grid energy function will be defect-corrected below, we can actually replace it by any convenient function $\bar{E}^c(u^c)$, as long as $\bar{E}^c(u^c)$ approximates $E^c(u^c)$ for smooth u^c . . . . For any given fine-grid configuration $\tilde{u}^f$ we then define the corrected interpolation $\tilde{I}_c^f$ (the FAS interpolation), given by $\tilde{I}_c^f u^c = \tilde{u}^f + I_c^f(u^c - I_f^c \tilde{u}^f)$ and the corrected coarse-grid energy $\tilde{E}^c(u^c) = \bar{E}^c(u^c) + (u^c, \tau_f^c)$, where $(\ ,\ )$, is the inner product and

$$\tau_f^c = (I_c^f)^T \nabla^f E^f(\tilde{u}^f) - \nabla^c \bar{E}^c(I_f^c \tilde{u}^f) ,$$

$(I_c^f)^T$ denoting the transpose of I_c^f." (Ref. 120, Sec. 7.1.)

[122]What follows is a slight modification (to treat the finite-temperature case) of the reasoning in Sec. 4 of their paper (Ref. 120).

[123]Brandt *et al.* do make a rather vague proposal for how to apply their idea to the finite-temperature case (Ref. 120, Sec. 7.2), but we are unable to make any precise sense of it.

[124]R. Gupta, Los Alamos National Laboratory Report No. LA-UR-87-2464, 1987 (unpublished), especially Sec. 5.7.

[125]K. Decker, Phys. Lett. B **210**, 207 (1988).

[126]K. E. Schmidt, Phys. Rev. Lett. **51**, 2175 (1983); M. Faas and H. J. Hilhorst, Physica **A135**, 571 (1986).

[127]R. C. Brower, R. Giles, K. J. M. Moriarty, and P. Tamayo, J. Comput. Phys. **80**, 472 (1989). Section 2 discusses algorithms for one-dimensional models, using the exact renormalized Hamiltonian; these ideas are essentially those of Schmidt (Ref. 126). Sections 3.1 and 3.3 give some ideas for "renormalization-group multigrid" (RGM) algorithms in higher dimensions, using *approximate* renormalized Hamiltonians; but we have been unable to understand precisely what is being proposed, and why the algorithm leaves invariant the correct Gibbs measure.

[128]J. Schröder, U. Trottenberg, and K. Witsch, in *Numerical Treatment of Differential Equations* (Ref. 29); U. Trottenberg, in *Computers, Fast Elliptic Solvers, and Applications,* Karlsruhe Conference, 1977, edited by U. Schumann (Advance, London, 1978). Useful background information can be found in review articles by F. W. Dorr, SIAM Rev. **12**, 248 (1970); **17**, 412 (1975); and by U. Schumann, in *Computers, Fast Elliptic Solvers, and Applications* (this reference).

[129]A. George, SIAM J. Numerical Anal. **10**, 345 (1973); A. J. Hoffman, M. S. Martin, and D. J. Rose, *ibid.* **10**, 364 (1973); R. J. Lipton, D. J. Rose, and R. E. Tarjan, *ibid.* **16**, 346 (1979); A. George and J. W. H. Liu, *Computer Solution of Large Sparse Positive Definite Systems* (Prentice-Hall, Englewood Cliffs, NJ, 1981).

[130]E. Dagotto and J. B. Kogut, Phys. Rev. Lett. **58**, 299 (1987); see Fig. 3.

[131]E. Dagotto and J. B. Kogut, Nucl. Phys. **B290** [FS20], 451 (1987).

[132]P. J. Rossky, J. D. Doll, and H. L. Friedman, J. Chem. Phys. **69**, 4628 (1978); R. T. Scalettar, D. J. Scalapino, and R. L. Sugar, Phys. Rev. B **34**, 7911 (1986); M. Creutz, Phys. Rev. D **38**, 1228 (1988).

[133]S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, Phys. Lett. B **195**, 216 (1987); M. Creutz, Phys. Rev. D **38**, 1228 (1988); H. Gausterer and S. Sanielevici, *ibid.* **38**, 1220 (1988); R. Gupta, G. W. Kilcup and S. R. Sharpe, *ibid.* **38**, 1278 (1988); A. D. Kennedy, in *Lattice 88,* proceedings of the 1988 Symposium on Lattice Field Theory, Fermilab, 1988, edited by W. A. Bardeen *et al.* (Nucl. Phys. B, Proc. Suppl.) (North-Holland, Amsterdam, 1989).

[134]Note that this is a partial-resampling algorithm, in the sense discussed in Sec. VII.

[135]A. D. Sokal, in *Computer Simulation Studies in Condensed-Matter Physics: Recent Developments,* edited by D. P. Landau and H.-B. Schüttler (Springer, Berlin/Heidelerg, 1988); R. G. Edwards and A. D. Sokal (in preparation).

[136]A. Ferrenberg (private communication).

[137]A more pessimistic (and probably more likely) scenario is that τ_{combined} behaves roughly like $\min(\tau_{\text{BT}}, \tau_{\text{MGMC}})$. Taking $\tau_{\text{BT}} \sim \lambda^{-q} L^{z_{\text{BW}}}$ and $\tau_{\text{MGMC}} \sim \lambda^p L^{z_{\text{HB}}}$ [note that (9.13) predicts $p = z_{\text{HB}}$], we find $\lambda_{\text{worst}} \sim L^{-(z_{\text{HB}} - z_{\text{SW}})/(p+q)}$ and hence predict that

$$\tau_{\text{combined}}(\lambda, L) \sim L^{(pz_{\text{SW}} + qz_{\text{HB}})/(p+q)} F(\lambda L^{(z_{\text{HB}} - z_{\text{SW}})/(p+q)})$$

for a suitable function F.

[138]M. Creutz, Phys. Rev. D **36**, 515 (1987); F. R. Brown and T. J. Woch, Phys. Rev. Lett. **58**, 2394 (1987); S. L. Adler, Phys. Rev. D **38**, 1349 (1988); U. M. Heller and H. Neuberger, *ibid.* **39**, 616 (1989); S. L. Adler, in *Lattice 88* (Ref. 133).

[139]H. Neuberger, Phys. Lett. B **207**, 461 (1988); Heller and Neuberger (Ref. 138).

[140]U. M. Heller and H. Neuberger, Phys. Lett. B **207**, 189 (1988); R. Gupta, J. DeLapp, G. G. Batrouni, G. C. Fox, C. F. Baillie, and J. Apostolakis, Phys. Rev. Lett. **61**, 1996 (1988).

[141]D. E. Knuth, *The Art of Computer Programming,* 2nd ed. (Addison-Wesley, Reading, MA, 1981), Vol. 2, Sec. 3.4.1, pp. 120 and 121; L. Devroye, *Non-Uniform Random Variate Generation* (Springer, New York, 1986), Sec. II.3.

[142]P. R. Tadikamalla, J. Am. Stat. Assoc. **75**, 683 (1980).

[143]M. Creutz and B. Freedman, Ann. Phys. **132**, 427 (1981); B. Freedman, P. Smolensky, and D. Weingarten, Phys. Lett. **113B**, 481 (1982); F. Cooper, B. Freedman, and D. Preston, Nucl. Phys. **B210** [FS6], 210 (1982); D. J. E. Callaway and R. Petronzio, *ibid.* **B240** [FS12], 577 (1984); A. D. Bruce, J. Phys. A **18**, L873 (1985); A. Milchev, D. W. Heermann, and K. Binder, J. Stat. Phys. **44**, 749 (1986).