DEUTSCHES ELEKTRONEN-SYNCHROTRON DESY

DESY 86-021
February 1986

LANGEVIN SIMULATIONS OF QCD, INCLUDING FERMIONS

by

A.S. Kronfeld
Deutsches Elektronen-Synchrotron DESY, Hamburg

ISSN 0418-9833

NOTKESTRASSE 85 · 2 HAMBURG 52

To be sure that your preprints are promptly included in the
HIGH ENERGY PHYSICS INDEX ,
send them to the following address (if possible by air mail) :

DESY
Bibliothek
Notkestrasse 85
2 Hamburg 52
Germany

DESY 86-021
February 1986

ISSN 0418-9833

LANGEVIN SIMULATIONS OF QCD, INCLUDING FERMIONS

Andreas S. Kronfeld

Deutsches Elektronen-Synchrotron DESY
Notkestraße 85, 2000 Hamburg 52
Federal Republic of Germany

INTRODUCTION

Perhaps the first question to ask is, "Why Langevin simulations?" One way of answering this question is to examine a disease from which most simulation algorithms suffer. The disease is critical slow down [1], and we encounter it in updating when $\xi/a \to \infty$ and in matrix inversion (needed to include fermions) when $m_q a \to 0$. A simulation that purports to solve QCD numerically will encounter these limits, so to face the challenge in the title of this workshop, we must cure the disease of critical slow down.

One can describe the disease in several ways. Let's focus on updating first; matrix inversions will be treated in detail below. Figure 1

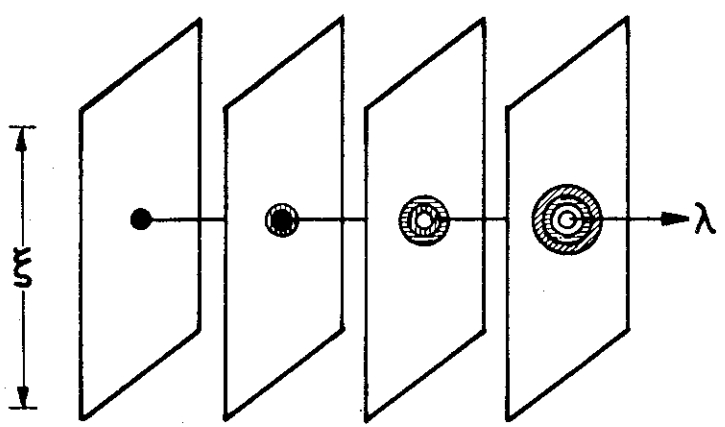

Fig. 1. Illustration of updating. When ξ is large, many updates are needed to propagate changes throughout the lattice.

illustrates the evolution of the system in λ = CPU time during the course of a typical simulation. Consider the effects of changes inside the black disc during the first update. Because the couplings are local, the only portion of the system sensitive to the change is in the circle with vertical stripes. During the second update the circle with the horizontal stripes feels the original changes, but only indirectly, through the changes to the vertically striped circle. As the updating process continues, more of the system feels the changes from the first update of the black circle, yet before one can consider a new configuration to be decorrelated from the original one, many updates are required. Indeed, for typical algorithms the number of updates needed grows as $(\xi/a)^2$. Physically, this critical slow down is due to the reluctance of changes at short distances to propagate to large distances. Numerically, the stability of an algorithm at short wavelengths requires a (moderately) small step size; critical slow down occurs when the effective long wavelength step size becomes tiny.

The remedy for this disease is an algorithm that propagates signals quickly throughout the system; i.e. one whose effective step size is not reduced for the long wavelength conponents of the fields. (Here the effective "step size" is essentially an inverse decorrelation time.) To do so one must resolve various wavelengths of the system and modify the dynamics (in CPU time) of the simulation so that all modes evolve at roughly the same rate. This can be achieved by introducing Fourier transforms. I will show how to implement Fourier acceleration for Langevin updating and for conjugate gradient matrix inversion. The crucial feature of these algorithms that lends them to Fourier acceleration is that they update the lattice globally; hence the Fourier transforms are computed once per sweep rather than once per hit.

SIMPLE LANGEVIN SIMULATIONS

For QCD the simplest Langevin algorithm is given by [2]:

$$U_{x,\mu}^{(\lambda+1)} = e^{-f\cdot T}\, U_{x,\mu}^{(\lambda)}$$

$$f_i = \sqrt{\epsilon_{ij}}\,\eta_j + \epsilon_{ij}\,[\partial_j S_g - \frac{1}{2}\,\mathrm{Re}\,(\xi^+ M^{-1}\partial_j M\,\xi)] \qquad (1)$$

where the T_a are antihermitean adjoint generators of SU(3). S_g is the pure gauge action and M is the fermion hopping matrix. The noise fields, η and ξ have zero mean and dispersion 2. The index i denotes group and position indices: $i = (a, x, \mu)$. Finally the field derivatives satisfy the Lie algebra: $[\partial_a, \partial_b] = -f_{abc}\,\partial_c$. The spatial dependence of ϵ_{ij} allows us to propagate the effects of $U_j^{(\lambda)}$ to all other links, $U_i^{(\lambda+1)}$, during one update.

The conceptual basis for Langevin updating is the stochastic quantization of Parisi and Wu [3]. When ϵ_{ij} is infinitesimal one can use the Fokker-Planck equation to show that, for large λ, the field U is distributed as desired. Unfortunately when ϵ_{ij} is finite the equilibrium action differs from the desired one by $O(\epsilon)$ terms. On the other hand, the nonlocality of ϵ_{ij} effects only details of the new terms. Define S $= S_g - \mathrm{Tr}\,\ln M$. Then

$$S = (1 + \frac{\bar{\epsilon}}{12} C_A)S + \frac{1}{4}\sum_{ij}\epsilon_{ij}\left\{2\partial_i\partial_j S - \partial_i S \partial_j S\right\} + \bar{\epsilon}\ (\text{fermionic headache}) \qquad (2)$$

where $\bar{\epsilon} = \mathrm{diag}(\epsilon_{ij})$. Later in the talk I will present algorithms whose leading corrections to the equilibrium action are $O(\epsilon^2)$, so please ignore the fermionic headache. Eq.(2) merely emphasizes that the $O(\epsilon)$ terms for nonlocal ϵ_{ij} are not radically different from those that appear when $\epsilon_{ij} = \bar{\epsilon}\delta_{ij}$.

FOURIER ACCELERATION OF UPDATING

The idea of Fourier acceleration is to introduce fast Fourier transforms (FFT's) when constructing the drift force f_i in Eq.(1). Then $\epsilon_{ij} \to \epsilon(p)$, and we pick $\epsilon(p)$ so that the decorrelation time, $N_{dc}(p)$, is the nearly same for all modes of the field. To illustrate how the FFT's can accelerate Langevin updating, consider a free scalar field. In momentum space the update rule reads

$$\phi^{(\lambda+1)}(p) = [1 - \epsilon(p)(p^2 + m^2)]\,\phi^{(\lambda)}(p) + \sqrt{\epsilon(p)}\,\eta^{(\lambda)}(p) \qquad (3)$$

where p and m are measured in lattice units. For $\phi^{(0)}(p) = 0$ Eq.(4) has a formal solution:

$$\phi^{(\lambda+1)}(p) = \sum_{\nu}[1 - \epsilon(p)(p^2 + m^2)]^{(\lambda-\nu)}\,\sqrt{\epsilon(p)}\,\eta^{(\nu)}(p) \qquad (4)$$

Then the correlations in λ are given by

$$\langle\phi^{(N+\lambda)}(p)\phi^{(\lambda)}(p)\rangle = \langle\phi^{(\lambda)}(p)\phi^{(\lambda)}(p)\rangle\,\exp[-N\epsilon(p)(p^2 + m^2)] \qquad (5)$$

so that the decorrelation time can be defined as

$$N_{dc}(p) = [\epsilon(p)(p^2 + m^2)]^{-1} \qquad (6)$$

If ϵ is local in coordinate space than $\epsilon(p)$ is independent of p. Then the decorrelation time for long wavelengths is $N_{dc}(0) \sim m^{-2} \sim (\xi/a)^2$. However, if one chooses $\epsilon(p) = \bar{\epsilon}/(p^2 + m^2)$, then $N_{dc}(p) = \frac{1}{\bar{\epsilon}}$ for all momenta.

For interacting theories one can pursue one of two strategies. If the dynamics is in or near a perturbative regime (as in QCD when $\beta \to \infty$) one can set $\epsilon(p)$ as above, albeit with a renormalized mass. Otherwise, one can study the correlations of Eq.(6) numerically as the simulation evolves; optimally,this will be done adaptively during the simulation. Both of these strategies have been successful in the XY model. In particular, the numerical determination of the optimal ϵ_{ij} has worked in the phase of the XY model with vortices.

Another illustrative example of how modifications to the Langevin dynamics can accelerate a simulation is given by a roundabout derivation of our fermion formulation. Instead of writing one Langevin equation for the gauge field, consider the equations derived from the action $S = S_g + S_f$ where $S_f = \phi^+ M^{-2}\phi$, which generates the two flavor theory [4]. The Langevin equation for the auxiliary scalar field is then [5]

$$\phi_i^{(\lambda+1)} = (\delta_{ik} - \epsilon_{ij} M_{jk}^{-2})\phi_k^{(\lambda)} + \sqrt{\epsilon}_{ij} \xi_j \qquad (7)$$

One wants to pick ϵ_{ij} so that the ϕ field decorrelates as quickly as possible. The obvious choice is $\epsilon_{ij} = \bar{\epsilon} M_{ij}^2$. Then detailed balance can be used to show that the equilibrium action is $S_f = \frac{2}{\sigma}(1 - \frac{1}{2}\bar{\epsilon})\, S_f$ exactly. By picking σ, the dispersion of ξ, properly one can also set $\bar{\epsilon} = 1$ so that the update rule becomes $\phi_i^{(\lambda+1)} = M_{ij}\xi_j$: the new ϕ is completely independent of the old, and S_f is simulated exactly! When this expression for ϕ is substituted into the gauge field Langevin equation, Eq.(1) results. As an added bonus one can then adjust the coefficient of the bilinear noise term to give any number of flavors: even, odd, or even fractional.

Gauge fields

Next, let's consider Fourier acceleration of the gauge field update. For several reasons one has to fix the gauge. The most physical reason is that short wavelength gauge artefacts can totally obscure our intuitive notion of momentum, and hence spoil the Fourier decomposition of the field into modes, some of which require acceleration. Another point of some importance is that a nonlocal ϵ without gauge fixing breaks gauge invariance; this could be solved by setting $\epsilon = \epsilon[U]$, but then the equilibrium is determined by the U dependence of ϵ as well as the U dependence of S. Consequently, we fix the gauge completely after each update. Numerical experience has shown that axial gauge gives unsatisfactory performance on an 8^4 lattice in the pure gauge theory; it is better to smooth the axial-gauge-fixed configurations by applying several iterations of a Landau gauge fixing. Details will be published elsewhere.

HIGHER ORDER ALGORITHMS

The problem with the algorithm presented in Eq.(1) is that $\bar{\epsilon}$ must be quite small when the procedure is Fourier accelerated. When ϵ is local, the leading corrections to the equilibrium action do not effect the continuum limit. (See Ref. 1). However, when ϵ is nonlocal this is no longer

true, so one needs a higher order differencing scheme for the Langevin equation. Such schemes generally require more memory. For systems without fermions the most efficient methods are similar to the Runge-Kutta algorithms for deterministic differential equations [6]. For simplicity I will leave off the indices needed for Fourier acceleration; they can be found in Ref.[7].

The Runge-Kutta tricks do not quite work for the bilinear noise term in Eq.(1) that introduces fermions, but Batrouni [8] has found a way around the problems. First one calculates a "tentative update" via the simple Euler rule:

$$\tilde{U}_{x,\mu} = e^{-\tilde{f}\cdot T}\, U_{x,\mu}^{(\lambda)}$$

$$\tilde{f}_i = \sqrt{\epsilon}\eta_i + \epsilon[\partial_i S_g - \frac{1}{4}\xi^+ A_i \xi] \qquad (8a)$$

followed by the final update, which has drift force

$$f_i = \frac{\epsilon}{2}(1 + C_A\frac{\epsilon}{6})[\partial_i S_g + \partial_i \tilde{S}_g - \frac{1}{4}\xi^+ A_i \xi - \frac{1}{4}\zeta^+ \tilde{A}_i \zeta] +$$

$$\sqrt{\epsilon}\eta_j[\delta_{ij} - \frac{1}{128}\mathrm{Re}(\xi^+ A_i \zeta \zeta^+ A_j \xi)] \qquad (8b)$$

In these equations $A_i = M^{-1}\partial_i M^2 M^{-1}$, ζ is an additional fermionic noise, and a tilde implies that S_g or A is evaluated using the tentative update, $\tilde{U}$.

An alternative procedure, which avoids the tentative update, requires a drift force with higher derivative of the action [7]:

$$U_{x,\mu}^{(\lambda+1)} = e^{-f\cdot T}\, U_{x,\mu}^{(\lambda)}$$

$$f_i = \epsilon(1 + C_A\frac{\epsilon}{24})[\partial_i S_g - \frac{1}{2}\mathrm{Re}(\xi^+ M^{-1}\partial_i M \xi)] +$$

$$\sqrt{\varepsilon}\eta_j \left\{ \delta_{ij} - \frac{1}{4} a_i \left[a_j S_g - \frac{1}{2} \mathrm{Re}(\xi^+ M^{-1} a_j M \xi) \right] \right.$$

$$\left. - \frac{1}{16} \mathrm{Re}\left[\xi^+ (M^{-1} a_i M a_j M M^{-1} + a_i M M^{-1}) \xi \right] \right\} \int \qquad (9)$$

Although this rule is not simple, it attains $O(\varepsilon^2)$ accuracy for the QCD with fermion loops with only two matrix inversions.

FOURIER ACCELERATION OF MATRIX INVERSION

To compute any of the drift forces in Eqs. (1), (9) or (10) one needs to solve the linear system of equations $M\psi = \xi$, where M is some lattice version of $\gamma_5(\not{D} + m_q)$; the γ_5 is included so that M is hermitean. We have used Wilson fermions. The matrix inversion is the most time consuming part of the update, so it is important to consider ways of accelerating routines like the conjugate gradient method [9]. Of course, matrix inversion is also important for quenched hadron spectroscopy.

The conjugate gradient method only works for positive definite matrices, so in practice we consider $M^2\psi = \phi = M\xi$. The convergence is governed by the ratio of the largest and smallest eigenvalues of M^2.

$$N_{CG} \approx \frac{|M^2|^{1/2}_{max}}{|M^2|^{1/2}_{min}} \approx \frac{1}{m_q a} \qquad (10)$$

since $M^2 \sim p^2 + m_q^2$. The gauge interaction produces terms that are off-diagonal in momentum space (the background gauge field exchanges momentum with the fermions), and gauge artefacts can obscure the p dependence of M^2.

We pre-condition M^2 by Fourier transforming (symbolized by $\hat{F}$):

$$\varepsilon(p)\hat{F}M^2\hat{F}^{-1}\hat{F}\psi = A\hat{F}\psi = \varepsilon(p)\hat{F}\phi \qquad (11)$$

The pre-conditioner $\varepsilon(p)$ is chosen so that the diagonal elements of A are all unity. In a smooth gauge we expect that

$$\frac{|A|^{1/2}_{max}}{|A|^{1/2}_{min}} \approx 1 \qquad (12)$$

As in the acceleration of updating there are two ways to determine the pre-conditioner. In a perturbative regime $\varepsilon(p) \sim (p^2 + m_q^2)^{-1}$, which explains the ε notation. Alternatively, one can compute the components of M^2 diagonal in momentum space and set $\varepsilon(p) = (M^{-2})_{pp}$. In fact, this approach gives a dramatic illustration of the practical necessity of fixing the gauge. Figure 2 shows the numerically determined $\varepsilon(p)$ for a propagator calculation in several gauges. All of the results are for SU(3) on a 8^4 lattice. Curve (a) shows the result for the free theory (all links are set equal to 1). Curve (b) shows $\varepsilon(p)$ for a $\beta = 5.8$ configuration without gauge fixing; evidently a wild gauge artefact has obscured any correlation between momentum and the diagonal elements of M^2.

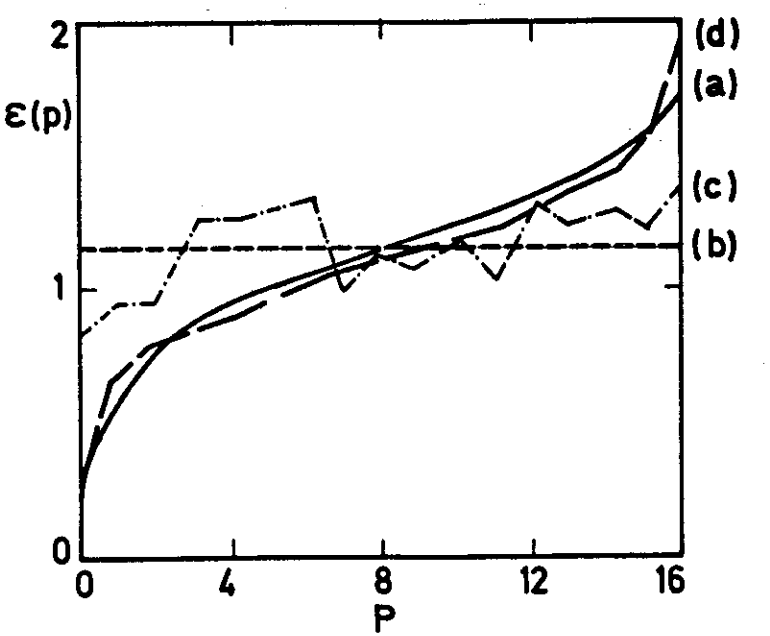

Fig. 2. Numerical computation of the pre-conditioner $\varepsilon(p)$. (a) Free field. (b) No gauge fixing. (c) Axial gauge. (d) Landau gauge.

Thus, although all diag(A) = 1, the spread of eigenvalues of A remains enormous. Fixing the same configuration to axial gauge improves this situation, as shown by curve (c), but not dramatically. However, Landau gauge, curve (d), produces a pre-conditioner that is qualitatively the same as for the free theory, as desired.

The acceleration provided by this technique is significant on an 8^4 lattice, and will become more significant on larger lattices. Figure 3 shows how many conjugate gradient iterations are needed to obtain convergence to fixed precision with and without Fourier acceleration.

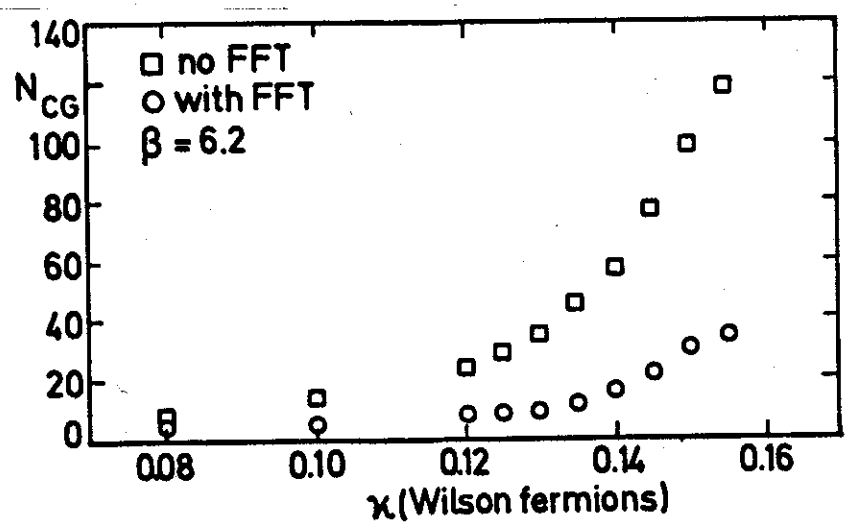

Fig. 3. Performance of the conjugate gradient with and without FFT's. As K approaches the critical value, the gain with FFT's is impressive.

Near physical values of the hopping parameter the accelerated algorithm needs 3 or 4 times fewer iterations. Owing to Eq.(11) this improvement factor should scale with the volume. The overhead for the FFT's is not serious: with everything except the FFT package optimized, a single iteration of the FFT conjugate gradient took only twice the CPU time of a normal iteration. Putting these factors together, we predict that the accelerated algorithm will be about 30 times faster for a 16^4 lattice; in practice, this means that smaller quark masses will be tractible.

CONCLUDING REMARKS

Let me summarize. Fourier acceleration is designed to reduce critical slow down. To make real progress towards the numerical solution of QCD we need algorithms for updating and matrix inversion that attack this problem. The FFT techniques are well suited to the attack, because they exploit our intuition (based on perturbation theory), and because they spread the effects of local changes throughout the system in the course of one iteration. Since the FFT's are inserted before and after each update, it is necessary to have an algorithm where "one update" means "one sweep," rather than "one hit." For matrix inversion this implies an algorithm such as the conjugate gradient, and for updating this implies Langevin or microcanonical [10] simulations. Fourier acceleration has been successful in simulating the XY model and in computing the quark propagator in quenched QCD. A complication when gauge fields are involved is the requirement of gauge fixing.

Finally, let us consider the prospects for a simulation of QCD (including vacuum polarization) on a $8^3 \times 16$ lattice. We expect Fourier acceleration to improve the matrix inversion by a factor of 4 and the updating by a factor of 16, so that the overall acceleration will be about a factor of 60. More generally, the number of operations needed for a simulation based on Eq.(1), or on the higher order generalizations, will be

$$N_{dc}(f_g I.^4 + f_f N_{CG} I.^4) \qquad (15)$$

where L is the linear size of the system in lattice units. Without Fourier acceleration both N_{dc} and N_{CG} grow dramatically as L increases, as indicated in Eqs.(2) and (11), respectively; this is critical slow down. The aim of our investigations is develop algorithms for which N_{dc} and N_{CG} are at worst slow functions of L. Then the simulation time will grow only as the volume of the system, which is unavoidable.

ACKNOWLEDGEMENTS

This talk summarizes the work of George Batrouni, Christine Davies, Garry Katz, Peter Lepage, Pietro Rossi, Ben Svetitsky, and Ken Wilson. It is a pleasure to recall our many interesting conversations, and I apologize if I have neglected their "pet projects" in preparing this talk. We have used a VAX 750 for program development and an FPS-164 array processor operating from an IBM 3081 host for production computing.

REFERENCES

1. G. Parisi in Progress in Gauge Field Theory, edited by G. 't Hooft, et al, (Plenum, New York, 1984.

2. G.G. Batrouni, G.R. Katz, A.S. Kronfeld, G.P. Lepage, B. Svetitsky, and K.G. Wilson, Phys. Rev. D32 (1985) 2736.

3. G. Parisi and Wu Y.-S., Sci. Sin. 24 (1981) 483

4. D.H. Weingarten and D.N. Petcher, Phys. Lett. 99B (1981) 333.

5. See also A. Ukawa and M. Fukugita, Phys. Rev. Lett. 55 (1985) 1854 for simulations based on this formula.

6. E. Helfand, The Bell System Technical Journal 58 (1979) 2289; H.S. Greenside and E. Helfand, ibid. 60 (1981) 1927.

7. A.S. Kronfeld, DESY report 86 - 006.

8. G.G. Batrouni, Cornell University report CLNS-85/665

9. M.R. Hestenes and E. Stiefel, J. Res. Natl. Bur. Stand. 49 (1952) 409.

10. J.B. Kogut, University of Illinois report ILL-(TH)-85-#75.