CHINESE JOURNAL OF PHYSICS VOL. 38, NO. 3-II JUNE 2000

Monte Carlo Methods for Quantum Field Theory

A. D. Kennedy

Maxwell Institute, The University of Edinburgh, UK.

(Received Mar. 14, 2000)

We present a review of some possibly interesting algorithmic techniques for the non-expert. We shall briefly survey a method for making Markov Chain Monte Carlo (MCMC) exact, the instabilities in symplectic integrators, exact noisy algorithms, Chebyshev (minimax) polynomial and rational approximations, and their relevance to dynamical GW fermions.

PACS. 11.15.Ha - Lattice gauge theory.
PACS. 11.30.Rd - Chiral symmetries.
PACS. 11.30.Fs - Global symmetries.

I. Introduction

The intention of this paper is to review some recent and a few not-so-recent advances in Monte Carlo simulation algorithms for lattice field theory with particular regard to their applicability to full QCD computations including Ginsparg–Wilson quarks.

The fundamental goals for the design of algorithms is the efficient and correct computation of quantities of physical interest; the mechanism for doing this need not itself necessarily be physically motivated. There are many techniques that can be applied to achieve the twin goals of reducing computational complexity[1] and of controlling systematic errors.

I-1. Reducing computational complexity

One important ingredient of reducing the algorithmic complexity of lattice QCD computations is the use of efficient methods for the solution of large sparse systems of linear equations. It is well known that Krylov space methods are more effective that simpler iterative schemes such as Jacobi's method. If we wish to implement Neuberger's operator

$$D_N \equiv \frac{1}{2}[1 + \gamma_5 \operatorname{sgn}(\gamma_5 D_W)] \tag{1}$$

and its inverse then just as in the case of computing D_W^{-1} we need to find good approximations that are cheap to compute, and ways of correcting for the residual errors. Indeed, if we want an effective algorithm for implementing Ginsparg–Wilson quarks then we need an approximation which can be made essentially arbitrarily accurate for a cost which grows only

[1] By "computational complexity" we mean the asymptotic scaling of the amount of computation with critical parameters such as the lattice spacing and the quark mass.

slowly. Since Neuberger's operator is constructed from the usual Wilson–Dirac operator D_W, and the only thing we can compute effectively with D_W is its application to a vector (i.e., a spinor field) we are naturally led to consider polynomial approximations to the sgn function. Since we are already familiar with the fact that Krylov space methods can apply D_W^{-1} efficiently, it follows that we are also able to use rational function approximations.

It is of course well known that any continuous function can be approximated arbitrarily well over a compact interval by a polynomial (the Weierstrass approximation theorem), and the proof of this result using Bernshteĭn polynomials gives an explicit construction of such an approximation. For a continuous function f over the unit interval

$$f_n(x) = \sum_{k=0}^{n} f\left(\frac{k}{n}\right) \left(\begin{array}{c} n \\ k \end{array} \right) x^k (1-x)^{n-k}$$

where the minimax error $\|f - f_n\|_\infty \equiv \sup_{x\in[0,1]} |f(x) - f_n(x)|$ goes to zero as $n \to \infty$. This approximation is not satisfactory, however, in the sense that in general the error falls only as a power of the degree n of the polynomial. One can find much better polynomial approximations for which the error falls exponentially with the degree. The optimal polynomials g_n which minimise $\|f - g_n\|_\infty$ are characterised by Chebyshev's theorem, which states that the pointwise error $|f(x) - g_n(x)| \leq \|f - g_n\|_\infty$ must take its maximum value exactly $n+2$ times over the unit interval. The best approximation to x^n over the interval $[-1, 1]$ by a polynomial of degree $n-1$ is easily seen to be $2^{-n+1}T_n(x) - x^n$ by this criterion, where the Chebyshev polynomials $T_n(x) \equiv \cos(n \cos^{-1} x)$ take their extreme values of ± 1 exactly $n+1$ times. It is immediately obvious that the minimax error thus falls as 2^{-n+1}. This result allows one to *economise* a series expansion of any continuous function f by truncating its expansion in terms of Chebyshev polynomials. If the original power series is only slowly convergent then this may not be good enough, so it usually preferable to approximate a fixed continuous function by solving Chebyshev's criterion numerically using a straightforward algorithm known as Remez' method.

Chebyshev's theorem also applies to rational function approximations, where the criterion is that $|f(x) - r_{n,d}(x)|$ must take its maximum value exactly $n + d + 2$ times over the unit interval, where n is the degree of the numerator of $r_{n,d}$ and d the degree of its denominator.[2] While there is no analogue of Chebyshev polynomials for rational function approximations, Remez' method is applicable in this case too. Just as in the case of polynomial approximations the minimax error usually falls exponentially in the degree, but for a given degree the minimax error is often many orders of magnitude smaller.

Of course, all this analysis only applies to continuous functions, and the alert reader will have no doubt noticed than sgn does not fall in this class. That is why it is desirable to extract the eigenvalues of D_W nearest to zero explicitly, e.g., by using a Ritz functional algorithm, and then to use a rational approximation to $\mathrm{sgn}(x)$ over the domain $[-1, -\epsilon] \cup [\epsilon, 1]$. This is equivalent to approximating $\mathrm{sgn}(x)$ by an odd function of x over the interval $[\epsilon, 1]$.

---

[2] It seems to always be the case that the "diagonal" rational approximations with $n = d$ are best.

I-2. Optimal amount of randomness in stochastic algorithms

There are two golden rules for designing efficient Monte Carlo algorithms:

1. Do not compute something exactly if you can produce a cheaper unbiased stochastic estimate, and
2. Do not compute something stochastically if you can afford to compute it exactly.

These rules may be summarised by saying that one should neither have too little nor too much randomness in the system. Of course, the art is in knowing which rule to apply in any given situation! The first rule is obvious, but the justification for the second is a little more subtle, and it is essentially that if you put too much randomness into a system it will undertake a random walk through the space of allowed states, and this causes the usual $\sqrt{N}$ critical slowing down. If you can arrange for the system to explore configuration space more systematically (but still stochastically with the correct weight) then one can avoid or at least reduce this critical slowing down.

A concrete example of these rules is provided by the Hybrid Monte Carlo algorithm as applied to QCD: the fermionic determinant is not computed deterministically, but is estimated with the introduction of stochastic pseudofermion fields. On the other hand, instead of making small global steps (as in the Langevin algorithm) or large local steps (as in link-by-link update schemes) it makes large global moves (trajectories through fictitious phase space of average length $\tau = \xi$ where ξ is the natural correlation length of the system under consideration. Naturally, the cost of each HMC trajectory (i.e., each Markov step) is large (of order $\xi/\delta\tau$, where the integration step size is $\delta\tau$), but this is more than offset by the reduction in the number of trajectories required to produce uncorrelated configurations. In the case of free field theory, where almost everything can be computed analytically, this means HMC has a cost which grows with the correlation length as ξ as opposed to ξ^2 for Langevin, and this leads to an overall volume dependence of the cost of $V^{5/4}$ as compared to $V^{4/3}$.

I-3. Control of systematic errors

There are many sources of systematic errors that need to have attention paid to them. In many cases these can be controlled by extrapolating to some limit in which the errors vanish, but this is rarely the most cost effective way of dealing with them.

I-3.1. Chiral limit

One of the principal errors which plague current lattice Monte Carlo computations is the fact that we cannot run at, or even near to, the physical values of the quark masses. This is for a variety of reasons: limitations on the physical lattice volume, the occurrence of "exceptional configurations" for which the quark kernel is not invertible, and most importantly the amount of computer time available. Whereas the staggered formulation discretisation allows us to probe closer to the chiral limit, it suffers from two fundamental difficulties: one is that it has the wrong number of flavours[3] (they only occur in multiples of four), and the other is that their "remnant of chiral symmetry" is not true chiral symmetry, and thus the computation of matrix elements is somewhat complicated. For Wilson fermions the major problem is that the theory is chiral only in the continuum limit, and then when the hopping parameter κ has been tuned to its critical value. This means that the chiral and continuum limits are inextricably interwoven, and this makes it inordinately expensive to probe the domain of physical quark masses. The introduction of Ginsparg–Wilson fermions gives us hope that we can separate the continuum and chiral limits of the theory by having a formulation with explicit chiral symmetry on the lattice, albeit only on mass-shell. If Ginsparg–Wilson fermions can be simulated effectively, and no unforeseen problems appear, then we should be optimistic that we will be able to carry out lattice QCD computations for far lighter quarks than are possible at present.

I-3.2. Discretisation errors

While Ginsparg–Wilson fermions hold promise of controlling the systematic errors associated with extrapolation to the chiral limit, we must not forget that there are other sources of systematic error which need to be controlled too. The most obvious of these are those due to the discretisation of the continuous manifolds of QCD in order to carry out numerical computations. These discretisations take three forms:

- The discretisation of the four dimensional space-time continuum onto a hypercubic lattice;
- The discretisation of "fictitious time" in Molecular Dynamics evolution required for the numerical integration of Hamilton's equations;
- The discretisation of the $SU(3)$ group manifold so that it can be represented in a computer. This latter is not often thought of as a discretisation error, but the use of finite precision floating point numbers to represent continuous real variables leads to systematic errors in just the same way as the other discretisations.

I-3.3. Lattices

It was found from the earliest days of computational lattice field theory that the obvious price of using a discrete hypercubic lattice — namely that the full Poincaré symmetry of the space-time continuum is broken to a discrete subgroup — appeared not to be too important in practice. Even on small coarse lattices full rotational symmetry appeared to be rapidly restored as the lattice spacing a was made small. Symanzik showed long ago [2, 3] that the effects of working at non-zero a could be removed order-by-order in a by introducing higher dimensional counterterms into the lattice action, and computed their coefficients for some simple models within perturbation theory. More recently this idea has attracted much attention after it was realised that the coefficients could also be computed

[3] We can, of course, take the "square root" of the staggered kernel, and indeed there are even methods for simulating this exactly [1]. However this square root is not a local action, and it is unclear whether it satisfies all the properties required of a physical quantum field theory.

non-perturbatively by numerical methods either by an ad hoc procedure [4, 5] or from first principles [6]. Even though the improvement coefficients are computed non-perturbatively, it is important to realise that they still only provide an asymptotic expansion in a, and thus even though the improvement can be made arbitrarily good for sufficiently small a the procedure breaks down if it is applied to very coarse lattices. There are those who doubt whether improved actions really give a net overall cost reduction for practical QCD computations.

I-3.4. Fictitious time

For any Molecular Dynamics type algorithm it is not possible to integrate the equations of motion exactly in closed form except in a few special cases [7, 8]. It is therefore necessary to break the classical trajectory into discrete time steps $\delta\tau$, which for small enough steps leads to errors in a trajectory of length τ of magnitude $\tau\delta\tau^{\alpha}$, where the exponent α can be made arbitrarily large by using high-order integration schemes [9, 10], although in practice the simplest lowest-order scheme is usually used because higher-order schemes tend to become unstable for smaller values of $\delta\tau$. This is discussed further in Section III-2.

The systematic errors caused by this integration time discretisation can be completely removed by adding a Metropolis accept/reject if a suitable reversible and area-preserving integration scheme is used. This is the Hybrid Monte Carlo algorithm [11], and it is in this sense that it is referred to as an "exact" algorithm. Symplectic integrators (q.v., Section III) provide a class of integration schemes with suitable properties. Of course, there is a price to pay for eliminating this source of systematic errors, and that is that when the errors in the approximate trajectory are large the corresponding Metropolis acceptance rate is very small.

I-3.5. Rounding errors

Computation over the field of real numbers is not only slow in practice, but is also undecidable in principle. Like all other large scale scientific computations lattice field theory uses *floating point* numbers as an approximation, in the hope that by starting with sufficient precision the accumulation of round-off errors will not affect at least the first few digits of the final result. To some extent this can be checked by carrying out the computation at two different precisions and verifying that the answers so obtained are compatible. The basic reason why this approach is feasible is that the hardware cost of n digit floating point arithmetic grows only as $\ln n$ for memory and $\ln n \ln\ln n$ for arithmetic operations, whereas rounding error propagation "normally" only consumes $\alpha \ln N$ digits where N is the number of sequential floating point operations performed[4]. The danger in practice is if the molecular dynamics of the system exhibits an exponential instability, in which case all significant digits are rapidly lost. Although the fundamental systematic errors are caused by the violation of the axioms of a mathematical field by floating point arithmetic it is the exponential amplification (which of itself is harmless) of these miniscule errors which can

[4] The coefficient α depends on the rounding mode used.

cause problems.

I-3.6. Chaos

The obvious cause of exponential amplification of rounding errors is that Hamilton's equations for the fictitious time molecular dynamics becomes chaotic. This means that the classical trajectories of two nearby points in fictitious phase space diverge exponentially with a characteristic *Liapunov exponent* ν. It was observed [12, 13] that the equations of motion for QCD (and for pure $SU(3)$ gauge theory) are intrinsically chaotic: this exponent was observed to scale with the correlation length and therefore seemed to be a property of the underlying continuum equations of motion [14]. However in reference [14] it was also noted that there was another source of chaos which occurred when the integrator became instable. For pure gauge theory and QCD with heavy quarks this only occurred for integration step sizes $\delta\tau$ much larger than the maximum allowed for a reasonable HMC acceptance rate, but similar measurements on UKQCD configurations with much larger lattices and lighter quarks [15] this instability seemed to set the maximum allowable step size. This will be discussed further in Section III-2.

While there is no evidence that 32-bit arithmetic does not suffice at present, there are some indications that higher precision will be required for computations with significantly lighter fermions. Of course, since the $SU(3)$ group elements lie in a compact manifold their components must be less than one in magnitude, and thus are most naturally represented as scaled integers. In this format they could be stored as 32-bit fractions, rather than the 24-bit fractional part[5] of 32-bit IEEE floating point numbers.

I-3.7. Convergence of iterative methods

For "inexact" fermionic Molecular Dynamics algorithms, such as HMD [16], there is another source of error. The computation of the fermionic "force" on the gauge fields requires the solution of a large set of simultaneous linear equations. Finding this solution is where most of the time in fermionic simulations is spent, and typically the solution is found using Krylov space methods such as Conjugate Gradients. The accuracy of such solutions $x = A^{-1}b$ is set by requiring the residual $\|Ax_n - b\|$ to be less than some prespecified value. Such criteria have to be applied with great care: for example, if the residual is computed directly from the three-term recurrence characterising CG and related methods then the *measured* residual can continue to fall even though the *actual* residual corresponding to the solution found does not.

For "exact" algorithms, such as HMC, there no systematic errors are caused by inaccurate solutions provided the solution is computed in a time-symmetric manner; an inaccurate solution will merely lead to a low acceptance rate. If the residual corresponding to the solution for the fermionic force is too large then this can lead to an integrator instability for any value of the integration step size.

---

[5] 26-bit if we are fair and count the hidden bit and the sign bit.

II. Markov chains

In the previous section we discussed a plethora of sources of systematic errors in lattice Monte Carlo computations. Underlying all the algorithms discussed there is some sort of Markov process, and we have been calling such algorithms "exact" if they have the desired distribution as a fixed point. If these methods are ergodic then they are guaranteed to converge to this fixed point whatever starting point is used, but both the "equilibration" time to reach this fixed point (to within statistical errors) and the subsequent autocorrelation time between effectively independent configurations can be very long. Furthermore, there is always the risk that some physically interesting observable (such as the topological susceptibility) couples to some particularly slow "mode" which otherwise evades diagnostics designed to test for equilibration. Recently methods have been devised for simple Markov chains which produce samples chosen from the equilibrium distribution with no such systematic errors. Although it is not (yet) known how to apply such methods to lattice QCD we shall outline the technique as it has great potential.

II-1. Coupling from the Past

The trick [17, 18] is to consider a Markov chain produced from some *fixed* sequence of random numbers[6] $x_1, x_2, \ldots$. For simplicity consider first a system with a finite state space $\{\phi_1, \ldots, \phi_n\}$ and a set of transition probabilities P_{ji}. The desired equilibrium distribution $Q = \sum_{i=1}^n Q_i \phi_i$ is a fixed point of this Markov matrix, $\sum_{i=1}^n P_{ji} Q_i = Q_j$. The resulting Markov chain is *deterministic* in that if the system is in state ϕ_i at step k it will go to state ϕ_j if and only if $x_k < P_{ji}$ (the exact formula depends on just how we use the random numbers, but we assume that some definite scheme has been chosen and is never changed).

In this situation the following "fly-paper principle" holds: if two chains coalesce they will stay together forever. This means that if the Markov process is ergodic then eventually all states will coalesce to some state ϕ_f with probability one. A corollary of this simple observation is that any state from time $-\infty$ will coalesce to ϕ_f, and thus ϕ_f is a sample from the fixed point distribution Q.

The naïve procedure of just following the chains from each state at time 0 until they coalesce to state ϕ_f at time t is invalid, because t is correlated to the sequence of random numbers used, but if instead one finds which state ϕ_f at time 0 was reached from time $-\infty$ then this bias is removed. This procedure of "coupling from the past" thus involves choosing some number of steps T, and then checking whether — for our fixed set of random numbers — all the states at time $-T$ have coalesced into some state ϕ_f at time 0. If they have not then one must increase T and try again with the same sequence of random numbers[7]

The resulting state ϕ_f is then a sample chosen from the distribution Q. Choosing a new sequence of random numbers and repeating the procedure then gives another

---

[6] Uniformly distributed over the unit interval.

[7] Each random number in the sequence is associated with some fixed time and this association is held fixed too.

completely independent equilibrium configuration.

II-2. Lattices

The time needed for all the Markov chains to coalesce is typically larger but of the same order of magnitude as the equilibration time of the underlying Markov process defined by P_{ji}. The major difficulty in implementing the "coupling from the past" algorithm is that the number of states is usually vast, and following them all explicitly is thus prohibitively expensive. Even for a 40 state one dimensional Ising model there are $2^{40} \approx 10^{12}$ states which the system could start from. One way of circumventing this difficulty is to find a *partial ordering* of the states for which

1. there is a largest and a smallest state (so the partial ordering forms a *lattice* in the mathematical sense), and
2. the ordering is preserved by each Markov step.

If such an ordering can be found then one merely needs to follow the Markov chain from the smallest and largest states at time $-T$, as when they have coalesced then a fortiori so will all the other states.

It is probably helpful to consider a simple example of how this works in practice. Consider a ferromagnetic Ising model with Hamiltonian $(J > 0)$

$$H(s) \equiv -J \sum_i \sum_{j \in \partial^* i} s_i s_j,$$

which we update using a local heatbath algorithm: that is, the new value for the spin at site i is set to $+1$ if

$$P(s_i) \equiv \frac{e^{H(s_i=+1)}}{e^{H(s_i=+1)} + e^{H(s_i=-1)}} < x_k$$

where x_k is the appropriate random number from the preset sequence. Consider now the ordering of Ising spin configurations defined by $s \geq s'$ iff $s_i \geq s'_i \; \forall i$: this satisfies the axiom that if $s \geq s'$ and $s' \geq s''$ then $s \geq s''$. Note that this is only a partial ordering, since $++-+--+ \geq ++----+$ but $++-+--+$ and $++--+-+$ do not have any definite ordering. Nevertheless, the totally ordered state with all spins $+1$ is greater than any other state, and the totally ordered state with all spins -1 is less than all others, so the partial ordering is a lattice. Most importantly the heatbath update preserves this ordering: if $s \geq s'$ and s'_i is updated to $+1$ then so is s_i, since[8] $P(s_i) \geq P(s'_i)$. The same random number x_k is used in the same way in both cases, which means that the partial order is preserved deterministically.

If we start far enough in the past from the two totally ordered states then the two Markov chains will eventually coalesce to some state ϕ_f by time zero (determined by our choice of random numbers), and this is then an equilibrium configuration.

---

[8] Let $\mu \equiv J \sum_{j \in \partial^* i} s_j$ and $\mu' \equiv J \sum_{j \in \partial^* i} s'_j$, then $s \geq s' \Rightarrow \mu \geq \mu' \Rightarrow e^{-2\mu'} \geq e^{-2\mu} \Rightarrow P(s_i) \geq P(s'_i)$.

Although suitable orderings are only known for a few Markov processes, they can be found for some continuous models, so there is no need to have a finite state space to use coupling from the past. If such an ordering — or some other practical means of finding when all the states have coalesced — could be found for lattice QCD then this would eliminate all systematic errors caused by lack of equilibration.

III. Symplectic integrators and instabilities

The next topic we address is how we can construct robust, reversible, and area-preserving discrete integration schemes for Molecular Dynamics evolution [19, 20]. The reason why this is possible is that the Hamiltonian is the sum of two terms, a *kinetic energy* term $T(p)$ which depends only upon the fictitious momenta p, and a *potential energy* term $S(q)$ which depends only upon the physical degrees of freedom q. Note that for HMC-like methods the potential of the fictitious Hamiltonian is just the action of the underlying field theory,

$$H(q,p) = T(p) + S(q) = \tfrac{1}{2}p^2 + S(q). \tag{2}$$

The generators of infinitesimal time translations for each of these terms separately are

$$Q \equiv T'(p)\frac{\partial}{\partial q} \qquad \text{and} \qquad P \equiv -S'(q)\frac{\partial}{\partial p},$$

and it is easy to integrate these terms separately for time τ

$$e^{\tau Q} : f(q,p) \mapsto f(q+\tau T'(p), p), \quad e^{\tau P} : f(q,p) \mapsto f(q, p-\tau S'(q)).$$

What we actually want to integrate is the full time evolution operator

$$\begin{aligned}\exp\left(\tau\frac{d}{dt}\right) &\equiv \exp\left(\tau\left\{\frac{dp}{dt}\frac{\partial}{\partial p} + \frac{dq}{dt}\frac{\partial}{\partial q}\right\}\right) = \exp\left(\tau\left\{-\frac{\partial H}{\partial q}\frac{\partial}{\partial p} + \frac{\partial H}{\partial p}\frac{\partial}{\partial q}\right\}\right)\\ &\equiv e^{\tau\hat{H}} = \exp\left(\tau\left\{-S'(q)\frac{\partial}{\partial p} + T'(p)\frac{\partial}{\partial q}\right\}\right).\end{aligned}$$

The way this can be built out of the components $e^{\tau Q}$ and $e^{\tau P}$ is by using the *Baker–Campbell–Hausdorff* formula, which tells us that $e^A e^B = e^{A+B+\delta}$ where the operator δ is in the free Lie algebra generated by $\{A, B\}$. Specifically, $\ln(e^A e^B) = \sum_{n\geq 1} c_n$ where $c_1 = A + B$ and c_{n+1} is

$$\frac{1}{n+1}\left\{-\tfrac{1}{2}[c_n, A-B] + \sum_{m=0}^{\lfloor n/2\rfloor}\frac{B_{2m}}{(2m)!}\sum_{\substack{k_1,\ldots,k_{2m}\geq 1\\ k_1+\cdots+k_{2m}=n}}[c_{k_1}, [\ldots, [c_{k_{2m}}, A+B]\ldots]]\right\},$$

with the B_m being Bernoulli numbers. The first few terms of this expansion are

$$\ln(e^A e^B) = \{A+B\} + \tfrac{1}{2}[A,B] + \tfrac{1}{12}\{[A,[A,B]] - [B,[A,B]]\} + \cdots. \tag{3}$$

In order to construct reversible mappings we use symmetric integrators: the following relation follows straightforwardly from the BCH formula (3)

$$\ln(e^{A/2}e^B e^{A/2}) = \{A+B\} - \tfrac{1}{24}\{2[B,[A,B]] + [A,[A,B]]\} + \cdots$$

Building an integrator out of a sequence of $\tau/\delta\tau$ steps, so that each commutator is associated with a factor of $\delta\tau$ which can be made sufficiently small to ensure convergence, we find that the PQP symmetric symplectic integrator is given by

$$
\begin{aligned}
U_0(\delta\tau)^{\tau/dt} &= \left(e^{\frac{1}{2}\delta\tau\, P} e^{\delta\tau\, Q} e^{\frac{1}{2}\delta\tau\, P}\right)^{\tau/\delta\tau} \\
&= \left(\exp\Big[(P+Q)\,\delta\tau - \tfrac{1}{24}\Big([P,[P,Q]] + 2[Q,[P,Q]]\Big)\,\delta\tau^3 + O(\delta\tau^5)\Big]\right)^{\tau/\delta\tau} \\
&= \exp\left[\tau\left((P+Q) - \tfrac{1}{24}\Big([P,[P,Q]] + 2[Q,[P,Q]]\Big)\,\delta\tau^2 + O(\delta\tau^4)\right)\right] \\
&\equiv e^{\tau\widehat{H'}} = e^{\tau(P+Q)} + O(\delta\tau^2).
\end{aligned}
$$

In addition to conserving energy to $O(\delta\tau^2)$ such symmetric symplectic integrators are manifestly area preserving and reversible. This integration scheme is usually referred to as *leapfrog* integration.

III-1. Conserved Hamiltonian

While such symplectic integrators only conserve the Hamiltonian (2) approximately, for every symplectic integrator there exists a Hamiltonian H' which is *exactly conserved.* For the PQP integrator we have

$$
\widehat{H'} \equiv \frac{\partial H'}{\partial p}\frac{\partial}{\partial q} - \frac{\partial H'}{\partial q}\frac{\partial}{\partial p} = \Big\{Q+P\Big\} - \tfrac{1}{24}\Big\{[P,[P,Q]] + 2[Q,[P,Q]]\Big\}\,\delta\tau^2 + O(\delta\tau^4).
$$

Substituting the explicit forms for the operators P and Q we obtain the vector field

$$
\begin{aligned}
\widehat{H'} &\equiv \frac{\partial H'}{\partial p}\frac{\partial}{\partial q} - \frac{\partial H'}{\partial q}\frac{\partial}{\partial p} \\
&= \left\{p\frac{\partial}{\partial q} - S'\frac{\partial}{\partial p}\right\} + \tfrac{1}{12}\left\{2pS''\frac{\partial}{\partial q} + (S'S'' - p^2S''')\frac{\partial}{\partial p}\right\}\delta\tau^2 + O(\delta\tau^4).
\end{aligned}
$$

Solving these differential equations for H' we find

$$
H' = H + \tfrac{1}{24}\left\{2p^2S'' - S'^2\right\}\,\delta\tau^2 + O(\delta\tau^4). \tag{4}
$$

Note that H' cannot be written as the sum of a p-dependent kinetic term and a q-dependent potential term, so we cannot expect to be able to find a simple symplectic integrator which conserves the Hamiltonian H exactly.

III-2. Instabilities

We have been somewhat cavalier about the convergence criteria, and indeed if the integration step size $\delta\tau$ is taken too large then the procedure becomes unstable. This is easy to understand, as it even occurs for free field theory [14]; indeed it suffices to consider a single harmonic oscillator with the Hamiltonian $\frac{1}{2}p^2 + \frac{1}{2}\omega^2q^2$, for which the leapfrog evolution operator is

$$
U_0(\delta\tau) = \begin{pmatrix} 1 - \frac{1}{2}(\omega\delta\tau)^2 & \omega\delta\tau \\ -\omega\delta\tau + \frac{1}{4}(\omega\delta\tau)^3 & 1 - \frac{1}{2}(\omega\delta\tau)^2 \end{pmatrix}.
$$

Since $\det U_0 = 1$ and $\operatorname{tr} U_0$ is real its eigenvalues must be of the form $\lambda_1 = e^{i\theta}, \lambda_2 = e^{-i\theta}$ or $\lambda_1 = \rho, \lambda_2 = 1/\rho$ with θ and ρ real. The latter situation corresponds to an instability, and occurs when the discriminant of the characteristic polynomial is positive, namely

$$\operatorname{disc}\det(U_0 - \lambda\mathbb{I}) = 4(\omega\delta\tau)^2 \left[\tfrac{1}{4}(\omega\delta\tau)^2 - 1\right] \geq 0.$$

An exponential instability therefore occurs for $\omega\delta\tau \geq 2$. An interesting question is what corresponds to ω for full QCD with light dynamical fermions? The evidence [15] indicates that it is the effective fermionic force on the gauge fields, which is a quantity that grows as the quark mass falls. For light quarks it is the instability in the integrator rather than the "perturbative" $\delta\tau$ errors which cause the method to fail (that is, lead to a negligible acceptance rate for HMC and wrong answers for HMC).

It is easy to find symmetric symplectic integrators whose errors are of arbitrarily high order in $\delta\tau$ using the method introduced by Campostrini [9, 10], but although the errors are much smaller as $\delta\tau \to 0$ the onset of instability also occurs for smaller values of $\delta\tau$, and if it is these instabilities rather than the HMC acceptance rate which limits the step size then ordinary leapfrog is preferable.

IV. Dynamical Ginsparg–Wilson fermions

How can these techniques be used for Ginsparg–Wilson computations? The use of polynomial and rational approximations to Neuberger's operator (1 have already been discussed in Section I-1. The major problem is that in order to compute propagators we need the inverse of D_N, and this involves an "outer" conjugate gradient iteration (the "inner" CG iteration being used to apply the rational function itself to a source vector).

There are some subtleties associated with using rational approximations. Consider as a simple but realistic example of rational approximation $r(x)$

$$\frac{1}{\sqrt{x}} \approx 0.3904603901 \frac{(x+2.3475661045)(x+0.1048344600)(x+0.0073063814)}{(x+0.4105999719)(x+0.0286165446)(x+0.0012779193)};$$

which is accurate to within almost 0.1% over the range $[0.003, 1]$. If we were to apply $r(D_W)$ to a vector in the most direct manner by applying the numerator and denominator sequentially then the latter linear system would have a large condition number. One the other hand, if we were to apply each denominator factor sequentially using three separate CG solutions then each solution would be much better conditioned. This intrinsic non-linearity of the performance of Krylov space algorithms is central to several ideas discussed at the Chiral '99 workshop.

One such idea, suggested by Neuberger and the SCRI group, is to use a partial fraction expansion of the rational function. This partial fraction expansion

$$0.3904603901 + \frac{0.0511093775}{x+0.0012779193} + \frac{0.1408286237}{x+0.0286165446} + \frac{0.5964845033}{x+0.4105999719}$$

appears to be numerically stable. We can use a multiple mass linear equation solver, thus reducing the cost significantly; essentially to the cost of the smallest "mass", which itself if larger than that of D_W itself.

For our example above computing the inverse of $r(D_W)$ is trivial — we just need to apply the inverse rational function. For the case of Neuberger's operator things are not so simple, as there are two non-commuting operators (γ_5 and D_W) involved. In this case we can use another trick suggested by Neuberger, namely to use a continued fraction expansion to reduce the nested CG problem to a single five dimensional CG solution. As a simple example (not involving continued fractions) let us start from a factorised rational approximation $\frac{A_1 A_2 A_3}{B_1 B_2 B_3} x = y$, and define some auxiliary variables by $x \equiv \xi_2$, $\frac{A_2}{B_2}\xi_2 \equiv \xi_1$, $\frac{A_1}{B_1}\xi_1 \equiv \xi_0$, and $\xi_0 = y$, or equivalently

$$\begin{pmatrix} y \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -B_1 & A_1 & 0 \\ 0 & -B_2 & A_2 \end{pmatrix} \begin{pmatrix} \xi_0 \\ \xi_1 \\ x \end{pmatrix}.$$

This five dimensional linear system can then be solved by a single Krylov space method. There are of course many ways of constructing such a five dimensional system, and how the condition number depends on the construction is not yet well understood. There are interesting connections between this five dimensional system (where the size of the fifth dimension is the degree of the denominator of the rational approximation) and the domain wall formulation of Ginsparg–Wilson fermions.

For dynamical fermions these same tricks can be used to compute the approximate RHMC force, but if we want an "exact" algorithm we need to construct a suitable noisy estimator for the Ginsparg–Wilson action. The Kennedy–Kuti algorithm or its variants cannot be applied directly, because Neuberger's operator does not have a series expansion about the origin, but some method of treating the eigenspace belonging to the low-lying eigenvalues separately might be possible.

Conclusions

We have discussed a selection of techniques which are interesting and may be useful for lattice QCD computations. In particular I hope we have reviewed some of the basic ideas underlying the RHMC algorithm, which is currently probably the most promising way of simulating with dynamical Ginsparg–Wilson quarks.

Acknowledgements

I would like to thank Ting-Wai Chiu, Kei-Fei Liu, and Lee Lin for organising a thoroughly stimulating and exciting workshop. I would also like to thank Ting-Wai for his great patience in waiting for my very belated contribution. Despite this I must apologise to the reader for having had to complete it in a hurry so that it is less polished and coherent than I would have wished. I would also like to thank the National Center for Theoretical Sciences (Hsinchu) for their kind hospitality and for providing a stimulating research environment.

References

[1] I. Horváth, A. D. Kennedy, and S. Sint, Nucl. Phys. B (Proc. Suppl.) **73**, 834 (1999).

[2] K. Symanzik, Nucl. Phys. **B226**, 187 (1983).

[3] K. Symanzik, Nucl. Phys. **B226**, 205 (1983).

[4] G. P. Lepage and P. B. Mackenzie, Phys. Rev. **D48**, 2250 (1993).

[5] M. Alford, W. Dimm, G. P. Lepage, G. Hockney, and P. B. Mackenzie, Phys. Lett. **B361**, 87 (1995).

[6] M. Lüscher, S. Sint, R. Sommer, P. Weisz, and U. Wolff, Nucl. Phys. **B491**, 323 (1997).

[7] A. D. Kennedy, Nucl. Phys. B (Proc. Suppl.) **30**, 96 (1993).

[8] A. D. Kennedy and K. M. Bitar, Nucl. Phys. B (Proc. Suppl.) **34**, 786 (1994).

[9] M. Campostrini and P. Rossi, Nucl. Phys. **B329**, 753 (1990).

[10] M. Creutz and A. Gocksch, Phys. Rev. Lett. **63**, 9 (1989).

[11] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, Phys. Lett. **195B**, 216 (1987).

[12] C. Liu, A. Jaster, and K. Jansen, Nucl. Phys. **B524**, 603 (1998).

[13] K. Jansen and C. Liu, Nucl. Phys. B (Proc. Suppl.) **53**, 974 (1997).

[14] R. G. Edwards, I. Horváth, and A. D. Kennedy, Nucl. Phys. **B484**, 375 (1997).

[15] B. Joó, A. D. Kennedy, and B. J. Pendleton, "Molecular Dynamics instabilities with light dynamical quarks," in preparation.

[16] R. T. Scalettar, D. J. Scalapino, and R. L. Sugar, Phys. Rev. **B34**, 7911 (1986).

[17] J. G. Propp and D. B. Wilson, in *Random Structures and Algorithms* **9**, 223 (1996).

[18] J. A. Fill, in *The Annals of Applied Probability* **8**, 131 (1998).

[19] I. Horváth, A. D. Kennedy, and S. Sint, "Analysis of inexact Monte Carlo algorithms," in preparation.

[20] A. D. Kennedy, "Monte Carlo algorithms and non-local actions.", to be published in *Monte Carlo Methods*, ed. N. Madras, (Fields Institute Communications Series).

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

[22] S. Duane, Nucl. Phys. **B257 [FS14]**, 652 (1985).

[23] S. Duane and J. B. Kogut, Phys. Rev. Lett. **55**, 2774 (1985).

[24] S. Duane and J. B. Kogut, Nucl. Phys. **B275**, 398 (1986).

[25] S. Gottlieb, W. Liu, D. Toussaint, R. L. Renken, and R. L. Sugar, Phys. Rev. **D35**, 2531 (1987).

[26] A. D. Kennedy and J. Kuti, Phys. Rev. Lett. **54**, 2473 (1985).

[27] M. Lüscher, Nucl. Phys. **B418**, 637 (1994).

[28] K. Jansen and R. Frezzotti, Phys. Lett. **B402**, 328 (1997).

[29] R. Frezzotti and K. Jansen, Nucl. Phys. **B555**, 395 (1999).

[30] R. Frezzotti and K. Jansen, Nucl. Phys. **B555**, 432 (1999).

[31] P. de Forcrand and T. Takaishi, Nucl. Phys. B (Proc. Suppl.) **53**, 968 (1997).

[32] P. de Forcrand, "Fermionic Monte Carlo algorithms for lattice QCD," hep-lat/9702009.

[33] I. Montvay, Nucl. Phys. **B466**, 259 (1996).

[34] I. Horváth, A. D. Kennedy, and S. Sint, "Monte Carlo algorithms for non-ultralocal actions II: PHMC and RHMC," in preparation.