Higher-Order Hybrid Monte Carlo Algorithms

Michael Creutz
Physics Department, Rutgers University, Piscataway, New Jersey 08855
and Physics Department, Brookhaven National Laboratory, Upton, New York 11973[a]

Andreas Gocksch
Physics Department, Brookhaven National Laboratory, Upton, New York 11973

(Received 14 April 1989)

We present a simple recursive iteration of the leapfrog discretization of Newton's equations which leads to a removal of the finite-step-size error to any desired order. This is done in a manner that preserves phase-space areas and reversibility, as required for use in the hybrid Monte Carlo method for simulating fermionic fields. The resulting asymptotic volume dependence is $V\exp[(\ln V)^{1/2}]$. We test the scheme on the (2+1)-dimensional Hubbard model.

PACS numbers: 05.50.+q

The past few years have seen an intense effort to develop better fermion Monte Carlo algorithms. It is probably fair to say that at this point in time algorithms based on the discretization of a differential equation, such as the original microcanonical algorithm,[1] the Langevin equation,[2] hybrids of the previous two,[3,4] and finally the hybrid Monte Carlo algorithm,[5] are the clear front runners and are the most widely used. We will refer to such algorithms as global algorithms since one step of the algorithm consists of an update of the entire lattice. Among the above algorithms the last one is unique in that it is also exact.

Any discretization scheme used for numerically integrating a differential equation will introduce finite-step-size errors. In this Letter we are concerned with the systematic cancellation of such errors. We have in mind mostly the hybrid Monte Carlo algorithm of Ref. 5, where a final acceptance step makes the algorithm satisfy detailed balance exactly. In this approach any reduction of the step-size error will help increase the final acceptance probability. However, much of what we say equally well applies to the approximate methods, where a smaller error amounts to smaller violations of detailed balance. Our presentation closely follows a previous work[6] by one of us, both in notation and philosophy. For another scheme for constructing higher-order algorithms we refer the reader to the paper by Kennedy.[7]

Let us quickly recall how the hybrid Monte Carlo algorithm works. We are interested in generating a probability distribution

$$P_{\mathrm{eq}}(A) \sim e^{-S(A)} \tag{1}$$

for a bosonic field $A(x)$ described by an action S. Fermionic fields are assumed to have been integrated out, giving an action which is slow to calculate; this favors global algorithms where S need be calculated only once per sweep of all lattice variables. Consider the classical Hamiltonian

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

and a discretization $T(\delta):(A,p)\rightarrow(A',p')$ of Hamilton's equations. The momentum p is a Gaussian random variable of unit width and δ represents the step size. Then, as long as $T(\delta)$ is both reversible in the sense

$$T^{-1}(\delta)=T(-\delta)\,, \tag{3}$$

and area preserving,

$$(dA,dp)=(dA',dp')\,, \tag{4}$$

simply accepting a step of the entire lattice under T with probability

$$P_{\mathrm{acc}}=\min[1,e^{H(p,A)-H(p',A')}]\,, \tag{5}$$

will exactly satisfy detailed balance and hence the system will approach the desired probability distribution. After each accept or reject step the momenta are refreshed with new Gaussian random variables. Since the algorithm is a global one, for large systems the change in energy (5) will be proportional to the volume, resulting in a possibly very small acceptance rate. Were we able to follow the classical trajectory *exactly* the acceptance would be unity because of energy conservation. We are therefore led to search for accurate discretizations of the equations of motion while maintaining the reversible and area-preserving properties. In addition, we require as little overhead as possible over the standard "leapfrog" approach. In particular when fermions are involved we want to keep the number of matrix inversions to a minimum.

To prepare for later developments we begin by rederiving the leapfrog method in a way which can be generalized. Consider the two transformations on phase space

$$T_A(\delta):(A,p)\rightarrow(A+p\delta,p) \tag{6a}$$

and

$$T_p(\delta):(A,p)\rightarrow(A,p-S'(A)\delta)\,. \qquad (6b)$$

Here $-S'(A)$ is the classical force. Both transformations are reversible and area preserving in the sense of Eq. (3) and Eq. (4); indeed, they are canonical transformations[8] with generating functions $F_2=p'A+p'^2/2$ and $p'A-S(A)$, respectively. The combined transformation T_1T_2 is *not* reversible in the sense of Eq. (3) since

$$\begin{aligned}(T_1T_2)^{-1}&=T_2^{-1}T_1^{-1}\\&=T_2(-\delta)T_1(-\delta)\neq T_1(-\delta)T_2(-\delta)\,.\end{aligned}$$

This is easily rectified by symmetrizing the product; for example, a discretization which *is* reversible is

$$T_\delta=T_p(\delta/2)T_A(\delta)T_p(\delta/2)\,. \qquad (7)$$

This is the leapfrog discretization which correctly reproduces the $O(\delta^2)$ terms in the equations of motion.

The full hybrid Monte Carlo algorithm is implemented with a final accept or reject step after a trajectory of N_{mic} leapfrog steps. This trajectory corresponds to an initial half step in p, then N_{mic} steps in A intercleaved with $N_{\text{mic}}-1$ steps in p, and finally another half step in p. The arguments of Refs. 6 and 9 indicate an asymptotic behavior with the volume $\sim V^{5/4}$ for this algorithm.

An immediate question that comes to mind is the following: Is there a composition of a number of T_A and T_p transformations which besides being reversible also correctly gives higher-order terms in the equation of motion? Such a transformation valid through order δ^4 was recently found by Campostrini.[10] His result inspired the present investigation and is a special case of our general construction.

In classical mechanics the Hamiltonian is the generator of translations in time, i.e., it generates motion along a classical trajectory. For any function $F(t)$ which depends on the phase-space variables at time t we can write

$$\begin{aligned}e^{H\delta}:F(t)\rightarrow F(t+\delta)&=F(t)+\{H,F\}\delta\\&\quad+\tfrac{1}{2}\{H,\{H,F\}\}\delta^2+\cdots\,. \qquad (8)\end{aligned}$$

The bracket denotes the unusual Poisson bracket. Suppose we have a transformation $T_n(\delta)$ giving a reversible, area-preserving discretization of Hamilton's equations accurate to some order n in δ. Thus we write for this transformation,

$$e^{H\delta}=T_n(\delta)+\Delta\delta^{n+1}+(\text{higher-order terms})\,. \qquad (9)$$

Here, addition of transformations is defined in the natural way in terms of what they do to points in phase space. We now consider combining two such transformations,

$$\begin{aligned}T_n(\delta_2)T_n(\delta_1)=e^{H(\delta_1+\delta_2)}&+\Delta e^{H\delta_1}\delta_2^{n+1}\\&+e^{H\delta_2}\Delta\delta_1^{n+1}+\cdots\,. \qquad (10)\end{aligned}$$

Since we are only interested in the leading error, we set δ_1 and δ_2 to 0 in the last two exponentials, thus obtaining

$$T_n(\delta_2)T_n(\delta_1)=e^{H(\delta_1+\delta_2)}+\Delta(\delta_2^{n+1}+\delta_1^{n+1})+\cdots\,. \qquad (11)$$

This relation has an important consequence when T is reversible in the sense of Eq. (3). In this case $T(-\delta)T(\delta)=1$ is exact. With Eq. (11) this implies that Δ must vanish if n is odd. Indeed, this is a simple way to see that the leapfrog algorithm errors only start at the third power of the step size. In going to higher-order schemes, as long as we maintain reversibility, the leading error in the equations of motion for A and p must be odd. The same is also true for the change in energy.

We are not ready for the announced higher-order scheme. Just as the ordinary leapfrog algorithm was built up of elementary canonical transformations the general transformation will be built up of elementary leapfrog steps. We proceed by induction, defining the transformation at any level in terms of the previous level, starting with the leapfrog. Assume a transformation $T_n(\delta)$ is a reversible, area-preserving discretization of Hamilton's equations accurate to even order $n=2m$, as in Eq. (9). Picking an arbitrary integer i, we consider taking i steps of size δ forward with this transformation, then one step backward with a size $s\delta=(2i)^{1/(n+1)}\delta$, finally followed with an additional i steps of size δ forward. Then Eq. (11) implies

$$\begin{aligned}T_{n+2}((2i-s)\delta)&=T_n(\delta)^iT_n(-s\delta)T_n(\delta)^i\\&=e^{H(2i-s)\delta+0\times\delta^{n+1}+\cdots} \qquad (12)\end{aligned}$$

will give an evolution accurate to order $n+2$. The fact that n increases by 2 is due to the symmetrization which keeps reversibility. Iterating this scheme recursively produces a discretization of the equations of motion to any desired order. The recursion starts from $m=1$ using the ordinary leapfrog. Campostrini's solution corresponds to the case $m=2$, $i=1$, $s=2^{1/3}$ with an appropriate rescaling of δ. Keeping i constant at each level gives a higher-order scheme involving $(2i+1)^{m-1}$ elementary leapfrog steps.

The algorithm is not unique; in particular, there is the parameter i giving the number of initial forward steps. While ideally this variable should be determined empirically at each level to optimize computer time, a simple criterion based on minimizing the largest step taken in a microcanonical trajectory can be given. Working at level m, we consider $i(j)$ to be the parameter i taken at level j, where $j\leq m$. We consider N_{mic} steps at the highest level. Calling δ the net step for the initial leapfrog, then the total trajectory length is

$$d=N_{\text{mic}}\delta\prod_{j=2}^{m}\{2i(j)-[2i(j)]^{1/(2j-1)}\}\,. \qquad (13)$$

For this trajectory the total number of leapfrog steps is

$$N_s = N_{\mathrm{mic}} \prod_{j=2}^{m} [2i(j)+1] , \tag{14}$$

and the maximum size for a single step is

$$s_{\max} = \delta \prod_{j=2}^{m} [2i(j)]^{1/(2j-1)}$$

$$= \frac{d}{N_s} \prod_{j=2}^{m} \left(\frac{2i(j)+1}{[2i(j)]^{(2j-2)/(2j-1)} - 1} \right) . \tag{15}$$

This suggests that at level j we take i to minimize $(2i+1)/[(2i)^{(2j-2)/(2j-1)}-1]$. Up to $j=9$, as well as asymptotically in j, the minimum occurs at $i=2j$.

We now discuss the asymptotic behavior as the volume becomes large. Here, and in the following, we will for simplicity assume that i is a constant. The analysis parallels that in Refs. 6 and 9. We have in mind fermionic problems where most of the time is spent in matrix inversion. We consider a trajectory of N_{mic} steps at level m before refreshing the momenta; thus, the computer time T grows as

$$T \sim V(2i+1)^{m-1} N_{\mathrm{mic}} . \tag{16}$$

This assumes a fixed number of inversion iterations per leapfrog step. The error per degree of freedom after N_{mic} steps should scale roughly as $\Delta = N_{\mathrm{mic}} \delta^{2m+1}$. The acceptance probability will fall[6] like $P_{\mathrm{acc}} \sim e^{-V\Delta^2}$ and hence we need

$$V N_{\mathrm{mic}}^2 \delta^{4m+2} \approx 1 . \tag{17}$$

Furthermore, we also want the total trajectory to have length of order 1 which yields

$$N_{\mathrm{mic}} \delta \approx 1 . \tag{18}$$

Eliminating δ from these relations, we see that N_{mic} should scale roughly as $N_{\mathrm{mic}} \sim V^{1/4m}$. In Eq. (16) this gives $T \sim V^{1+1/4m}(2i+1)^{m-1}$. Minimizing T with respect to m leads to $m \sim (\ln V)^{1/2}$ so that $T \sim V \exp[(\ln V)^{1/2}]$. Hence if we adjust m appropriately as the volume of our system increases, the correction to a linear dependence grows slower than any power of the volume.

As a first check on our method we looked at the deviation of the energy when it was applied to the simple classical anharmonic oscillator. We found the expected dependence of the error on the step size as a function of m as long as $\delta \leq 1$. As a real test of the method we applied it to a many-fermion system, the two-dimensional Hubbard model. The quantum Hamiltonian is

$$H = -K \sum_{i,j,\sigma} a_{i\sigma}^{\dagger} a_{j\sigma} - \tfrac{1}{2} U \sum_{i} (a_{i\uparrow}^{\dagger} a_{i\uparrow} - a_{i\downarrow}^{\dagger} a_{i\downarrow})^2 , \tag{19}$$

where the σ sum runs over spin up and down and the a's satisfy canonical anticommutation relations. The system is simulated by the hybrid Monte Carlo algorithm after rewriting the partition function $Z = \mathrm{tr}\, e^{-\beta H}$ as a path integral. All the details of how this is done can be found in Ref. 6. The only difference with the simulation here is that two separate auxiliary fields were used, one for each spin, rather than the single one used in Ref. 6. As we said before, implementing our higher-order scheme only required minimal changes to the program. In our simulations we always chose a value $K=1$ for the hopping parameter and $U=2$ for the strength of the four-fermion interaction, and finally we kept the inverse temperature at $\beta=1$.

TABLE I. The acceptance at the end of a trajectory of length 1 for various algorithms. Averages are taken over ten trajectories.

V	m	i	N_{mic}	δ	Acceptance
$12^2 \times 8$	1	1	15	0.067	0.95(01)
$12^2 \times 8$	2	2	3	0.138	0.68(11)
$12^2 \times 8$	2	3	2	0.119	0.47(10)
$12^2 \times 8$	2	7	1	0.086	0.65(11)
$20^2 \times 8$	1	1	15	0.067	0.95(03)
$20^2 \times 8$	2	2	3	0.138	0.51(11)
$20^2 \times 8$	2	3	2	0.119	0.42(13)
$20^2 \times 8$	2	7	1	0.086	0.27(10)
$12^2 \times 8$	1	1	8	0.125	0.81(04)
$20^2 \times 8$	1	1	8	0.125	0.82(08)

In order to compare the different algorithms (different m and i) we used the following simple procedure. At a given fixed volume we chose N_{mic}, m, and i such that the computer time was roughly a constant. Subsequently, we determined δ to give a total trajectory length of exactly unity. All other things being equal, the best algorithm is clearly the one with the highest acceptance after the N_{mic} steps. Quite arbitrarily we chose for the standard leapfrog algorithm $N_{\mathrm{mic}}=15$ and $\delta=\frac{1}{15}$. Hence in all the other cases we insisted that the number of elementary leapfrog steps was about 15, leading to the same amount of computer time required to travel a distance 1. Our results are summarized in Table I.

It is evident that on these moderate-size lattices the standard leapfrog method is hard to beat. The problem with the higher-order algorithms seems to correlate with the large backwards step in the middle of the algorithm. Note the big fluctuations in the acceptance. As a comparison, we have also included the leapfrog algorithm with a larger step size in the last rows of the table. On larger volumes one must presumably go to larger values of N_{mic} in the leapfrog method. This will force simulations to smaller δ values than used here. In that case the extra two powers gained may help the acceptance.

To summarize, we have presented a simple recursive method to generate higher-order global Monte Carlo algorithms from simple leapfrog steps. Using a reentrant subprogram (this is particularly amusing to try in FORTRAN) one can easily modify a hybrid Monte Carlo program to work to arbitrary order. There are obvious variations on our general scheme that we have not discussed. For example, one may want to try additional accept or reject steps on a fraction of the variables after each of the N_{mic} iterations in order to increase the final acceptance. The optimum parameters are presumably model dependent.

We would like to thank M. Campostrini for communicating his results to us prior to publication. This manuscript has been authored under Contract No. DE-AC02-76-CH00016 with the U.S. Department of Energy.

[a] Permanent address.

[1] J. Polonyi and H. W. Wyld, Phys. Rev. Lett. **51**, 2257 (1983).

[2] G. G. Batrouni, G. R. Katz, A. S. Kronfeld, G. P. Lepage, B. Svetitsky, and K. G. Wilson, Phys. Rev. D **32**, 2736 (1985).

[3] S. Duane and J. Kogut, Phys. Rev. Lett. **51**, 2257 (1983).

[4] S. Gottlieb, W. Liu, D. Toussaint, and R. L. Sugar, Phys. Rev. D **35**, 2611 (1986).

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

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

[7] A. D. Kennedy, Supercomputer Computations Research Institute, Florida State University, Report No. FSU-SCRI-88-132, 1988 (to be published).

[8] H. Goldstein, *Classical Mechanics* (Addison-Wesley, Reading, MA, 1950).

[9] R. Gupta, G. Kilcup, and S. Sharp, Phys. Rev. D. **38**, 1278 (1988).

[10] M. Campostrini (private communication); M. Campostrini and P. Rossi (to be published).