Odd-flavor Simulations by the Hybrid Monte Carlo [a]

Tetsuya Takaishi
Hiroshima University of Economics, Hiroshima, 731-0192, JAPAN.
E-mail:tt-taka@hue.ac.jp

Philippe de Forcrand
CERN, Theory Division, CH-1211 Geneva 23, Switzerland
ETH-Hönggerberg, CH-8093 Zürich, Switzerland
E-mail:forcrand@itp.phys.ethz.ch

The standard hybrid Monte Carlo algorithm is known to simulate even flavors QCD only. Simulations of odd flavors QCD, however, can be also performed in the framework of the hybrid Monte Carlo algorithm where the inverse of the fermion matrix is approximated by a polynomial. In this exploratory study we perform three flavors QCD simulations. We make a comparison of the hybrid Monte Carlo algorithm and the R-algorithm which also simulates odd flavors systems but has step-size errors. We find that results from our hybrid Monte Carlo algorithm are in agreement with those from the R-algorithm obtained at very small step-size.

1 Introduction

Recent lattice QCD simulations include effects of dynamical fermions. The standard established algorithm to simulate dynamical QCD is the Hybrid Monte Carlo (HMC) algorithm [1] but it is limited to simulations of an even number of degenerate flavors. It would be desirable to simulate lattice QCD with three flavors since there exist three light quarks, i.e. u,d,s quarks, in the real world. Simulations with an odd number of flavors have been performed using the R-algorithm [2]. This algorithm, however, is not exact: it causes systematic errors of order $\Delta\tau^2$, where $\Delta\tau$ is the step-size of the Molecular Dynamics evolution. A careful extrapolation to zero step-size is therefore needed to obtain exact results. Nevertheless, it is common practice to forego this extrapolation and to perform simulations with a single step-size chosen small enough that the expected systematic errors are smaller than the statistical ones. We want to point out that there is an alternative to the R-algorithm, which gives arbitrarily accurate results without any extrapolation[3].

Some time ago, a local algorithm, the so-called "Multiboson algorithm", was proposed by Lüscher [4], in which the inverse of the fermion matrix is approximated by a suitable Chebyshev polynomial. Originally he proposed it for two flavors QCD. Boriçi and de Forcrand [5] noticed that the determinant of a

[a] Talk presented by T.Takaishi

fermion matrix can be written in a manifestly positive way using a polynomial approximation, so that one can simulate odd flavors QCD with the multiboson method. Indeed, using this method, one flavor QCD was simulated successfully [6]. The same polynomial approximation can be applied for the HMC algorithm [7], which means that the HMC algorithm also has the possibility of simulating odd flavors. Here we give the formulation for simulating odd flavors with the HMC algorithm and perform three flavors simulations. Then we compare our results with those of the R-algorithm.

2 Formulation

2.1 n_f=2

The lattice QCD partition function with n_f degenerate quark flavors is given by

$$Z = \int dU \det(D)^{n_f} \exp(-S_{gauge}), \tag{1}$$

where D is the fermion matrix and in this study we use Wilson fermions. For $n_f = 2$, the partition function is

$$Z = \int dU \det(D)^2 \exp(-S_{gauge}). \tag{2}$$

In the formulation of the HMC algorithm, $\det(D)^2$ is treated as

$$\det D^2 \sim \int d\phi^\dagger d\phi \exp(-\phi^\dagger D^{\dagger -1} D^\dagger \phi), \tag{3}$$

where the γ_5 hermiticity of the fermion matrix D, i.e. $D = \gamma_5 D^\dagger \gamma_5$, is used.

Introducing momenta P conjugate to the link variables U, the partition function is rewritten as

$$Z = \int dU dP \exp(-H), \tag{4}$$

where the Hamiltonian H is defined by

$$H = \frac{1}{2}P^2 + S_{gauge} + \phi^\dagger D^{\dagger -1} D^{-1} \phi. \tag{5}$$

This Hamiltonian is used for the Molecular Dynamics (MD) simulation of the standard HMC algorithm. Eq.(5) has a computational difficulty in MD simulations since one must solve $x = D^{-1}\phi$ type equations which in general

take a large amount of computational time for a large fermion matrix and/or for a small quark mass.

Following Lüscher[4], the inverse of D can be approximated by a polynomial:

$$1/D \approx P_n(D) \equiv \prod_{k=1}^{n} (D - Z_k) \tag{6}$$

where Z_k are roots of the polynomial $P_n(D)$:

$$Z_k = 1 - exp(i\frac{2\pi k}{n+1}). \tag{7}$$

The rate of convergence of the approximation depends on the quark mass (See Sec.3).

Replacing D^{-1} in eq.(5) by $P_n(D)$ we obtain an approximate Hamiltonian,

$$H_n = \frac{1}{2}P^2 + S_{gauge} + \phi^\dagger P_n(D)^\dagger P_n(D)\phi. \tag{8}$$

An advantage of using H_n is that no solver calculation is required in the MD evolution. Instead, one needs n multiplications by the matrix D. Originally H_n was introduced to reduce computational work. Indeed, it was shown that H_n can provide some gain over the standard HMC algorithm[7].

H_n does introduce some systematic errors from the polynomial approximation. For the n_f=even case, however, these errors are easily corrected at the Metropolis step by using the exact Hamiltonian of eq.(5)[7].

The domain of convergence of $P_n(D)$ is bounded by a circle centered at (1,0) which goes through the origin. If all eigenvalues of D fall inside this domain, $P_n(D)$ converges exponentially. Otherwise, $P_n(D)$ does not converge, which may happen for some exceptional configurations. Our algorithm rejects these configurations at the Metropolis step. This domain of convergence can be changed by adopting another approximating polynomial. However, the origin must be excluded. Together with connectedness and conjugate symmetry of the spectrum, this implies that the real negative axis is always excluded from the domain of convergence for any polynomial. Configurations with real negative Dirac eigenvalues will be rejected by our polynomial algorithm.

2.2 $n_f = 1$

In this case, we have to consider $\det D$. $\det D$ can not be expressed in a manifestly positive manner using the same treatment of eq.(3). Thus the standard HMC algorithm can not handle $n_f = 1$ or n_f=odd simulations.

The multiboson algorithm was originally developed for a simulation of n_f=2 QCD [4]. After invention of the multiboson algorithm, Boriçi and de Forcrand [5] noticed that a single $\det D$ can be treated in a manifestly positive way and an $n_f = 1$ multiboson simulation was performed to study thermodynamics of $n_f = 1$ QCD [6].

As before, the inverse of the fermion matrix D, using a polynomial of degree $2n$, is approximated as [4,5]

$$1/D \approx \prod_{k=1}^{2n} (D - Z_k), \tag{9}$$

where $Z_k = 1 - \exp(i\ 2\pi k/(2n+1))$. Noticing that the Z_k's come in complex conjugate pairs, eq.(9) is rewritten as

$$1/D \approx \prod_{k=1}^{n} (D - \bar{Z}_k)(D - Z_k). \tag{10}$$

Using the γ_5 hermiticity of the fermion matrix, we find that $\det(D - \bar{Z}_k) = \det(D - Z_k)^\dagger$. Thus the determinant of D is written as

$$\det(D) \sim \det(T_n^\dagger(D) T_n(D))^{-1}, \tag{11}$$

where $T_n(D) \equiv \prod_{k=1}^{n} (D - Z_k)$. Using an integral form of the determinant, we obtain

$$\det(D) \sim \int d\phi^\dagger d\phi \exp(-\phi^\dagger T_n^\dagger(D) T_n(D) \phi). \tag{12}$$

The term $\phi^\dagger T_n^\dagger(D) T_n(D) \phi$ is manifestly positive. Then we may define the Hamiltonian of $n_f = 1$ QCD as

$$H = \frac{1}{2} P^2 + S_{gauge} + \phi^\dagger T_n^\dagger(D) T_n(D) \phi. \tag{13}$$

With this Hamiltonian there is no difficulty to perform HMC algorithm. The domain of convergence of the approximation eq.(12) is the same as for $n_f = 2$. Exceptional configurations for which eigenvalues fall outside this domain will be rejected at the Metropolis step.

2.3 $n_f = 2 + 1$

The partition function of $n_f = 2 + 1$ QCD is given by

$$Z = \int dU \det \tilde{D}^2 \det D \exp(-S_{gauge}), \tag{14}$$

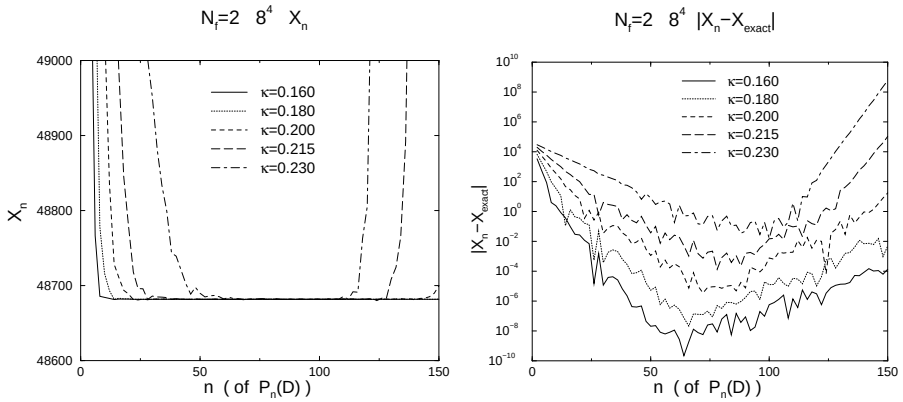

Figure 1: (left): X_n versus degree n. (right):$|X_n - X_{exact}|$ versus degree n.

where the notations $\tilde{D}$ and D are introduced to distinguish the two different quark masses. Using eq.(3) for $\det \tilde{D}^2$ and eq.(12) for $\det D$,

$$\det \tilde{D}^2 \det D \sim \int d\tilde{\phi}^\dagger d\tilde{\phi} d\phi^\dagger d\phi \exp(-\tilde{\phi}^\dagger \tilde{D}^{\dagger -1} \tilde{D}^{-1} \tilde{\phi} - \phi^\dagger T_n^\dagger(D) T_n(D) \phi). \quad (15)$$

We define $n_f = 2 + 1$ Hamiltonian by

$$H = \frac{1}{2} P^2 + S_g + \tilde{\phi}^\dagger \tilde{D}^{\dagger -1} \tilde{D}^{-1} \tilde{\phi} + \phi^\dagger T_n^\dagger(D) T_n(D) \phi. \quad (16)$$

Two remarks are in order: (i) the degree n of the approximating polynomial may be different during the Molecular Dynamics trajectory and for the Metropolis acceptance test; the former can be made arbitrarily small and tuned for maximum efficiency, while the latter should be taken very large to enforce the correct measure; (ii) the two bosonic fields ϕ and $\tilde{\phi}$ could be replaced by a single one, with action $\phi^\dagger T_n^\dagger(D) \tilde{D}^{\dagger -1} \tilde{D}^{-1} T_n(D) \phi$. For simplicity, in this exploratory study we use two distinct bosonic fields and a single degree for the approximating polynomial.

3 Convergence

3.1 $n_f = 2$

In order to see the rate of convergence of $P_n(D)$, we calculate the quantity $X_n = \phi^\dagger P_n^\dagger(D) P_n(D) \phi$. In the limit $n \to \infty$, X_n goes to $X_{exact} \equiv$

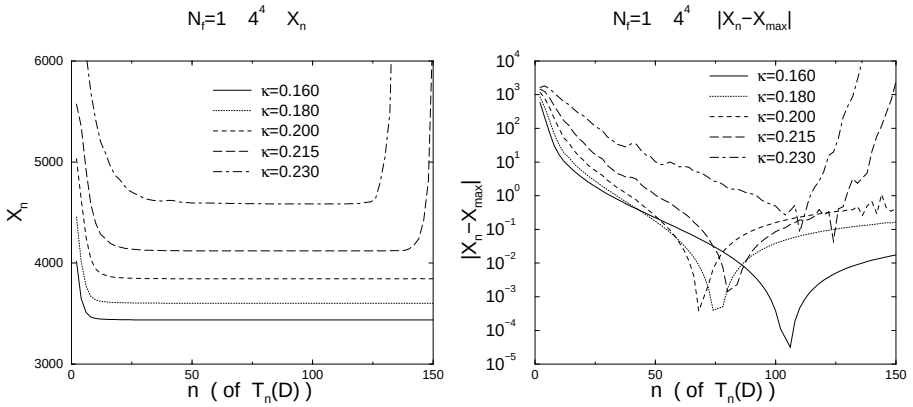

Figure 2: (left): X_n versus degree n. (right):$|X_n - X_{max}|$ versus degree n.

$\phi^\dagger {D^\dagger}^{-1} D^{-1}\phi$. First, we choose $X_{exact} = \eta^\dagger\eta$ where η is a random gaussian vector. Then the vector ϕ is set to $\phi \equiv D\eta$. The accuracy of X_n is measured by the difference between X_n and X_{exact}. We use a random gauge configuration for this analysis. Figure 1:(left) shows X_n versus the degree n for different quark masses. Here the same η is used for each calculation of X_n. X_n converges to one value as n increases, but at high degree n, X_n diverges, which can be understood due to the rounding errors of our computer, where calculations are performed with 64-bit accuracy.

Figure 1:(right) shows the accuracy of X_n by $|X_n - X_{exact}|$. Exponential convergence is seen for each quark mass, but the rate of convergence is slow for small quark masses.

3.2 $n_f = 1$

We do the same analysis as for $n_f = 2$, but for $n_f = 1$, the value of X_{exact} is not known. So we calculate the quantity $X_n = \phi^\dagger T_n^\dagger(D) T_n(D)\phi$, where the vector ϕ is a gaussian random vector, and we use a random gauge configuration. We assume that X_n goes to a certain value in the limit of $n \to \infty$.

Figure 2:(left) shows X_n as a function of degree n. X_n seems to converges to a certain value when the degree n increases. At high degree n, X_n diverges as in the case of $n_f = 2$.

To see the rate of convergence, we calculate $|X_n - X_{max}|$ where X_{max} is defined by $X_{max} = X_m$, $m >> n$. Due to the rounding errors, we can not

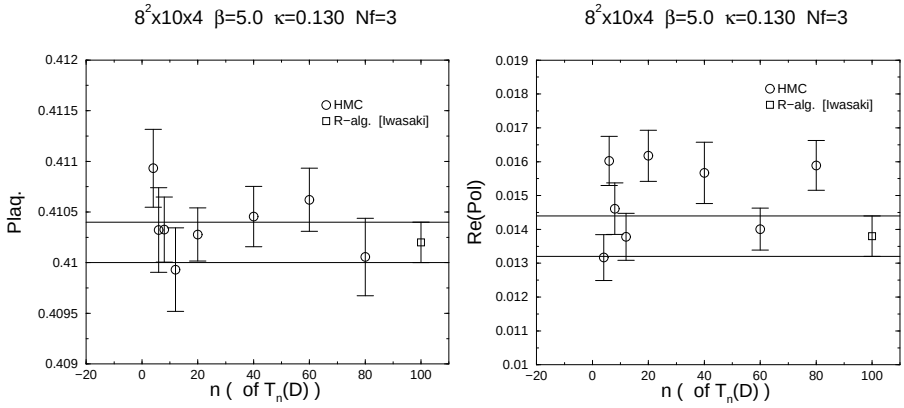

Figure 3: (left): Plaquette of $n_f = 3$ flavor QCD on an $8^2 \times 10 \times 4$ lattice at $\beta = 5.0$ and at $\kappa = 0.130$ as a function of degree n. (right): Real part of Polyakov loop.

take very large m. We take a maximum number m where the rounding errors still do not appear. Figure 2:(right) shows $|X_n - X_{max}|$ as a function of degree n. The dips seen in the figure are just due to the fact that at those points $X_n = X_{max} \equiv X_m$. The convergence seems to be exponential, but the rate of convergence is slow for small quark masses as in the $n_f = 2$ case.

4 Simulations

We perform simulations of three flavors QCD on an $8^2 \times 10 \times 4$ lattice at $\beta = 5.0$ with $\kappa = 0.130$ and 0.160. We measure the plaquette and Polyakov loop varying the degree n and compare them with those from the R-algorithm obtained with a step-size $\Delta\tau = 0.01$[8]. Figures 3 and 4:(left) show the plaquette as a function of n at $\kappa = 0.130$ and 0.160, respectively. Except for very small n, the results from the HMC algorithm agree with those from the R-algorithm within statistical errors. Results of the Polyakov loop are shown in figures 3 and 4:(right). Except for a small discrepancy seen in figure 3, the results from the HMC algorithm are in agreement with those from the R-algorithm. Note that convergence is not monotonic in n.

5 Conclusions

We formulated an odd-flavor HMC algorithm using a polynomial approximation. Simulations of three flavors QCD were performed. We found that the plaquette values are consistent with those from the R-algorithm at very small

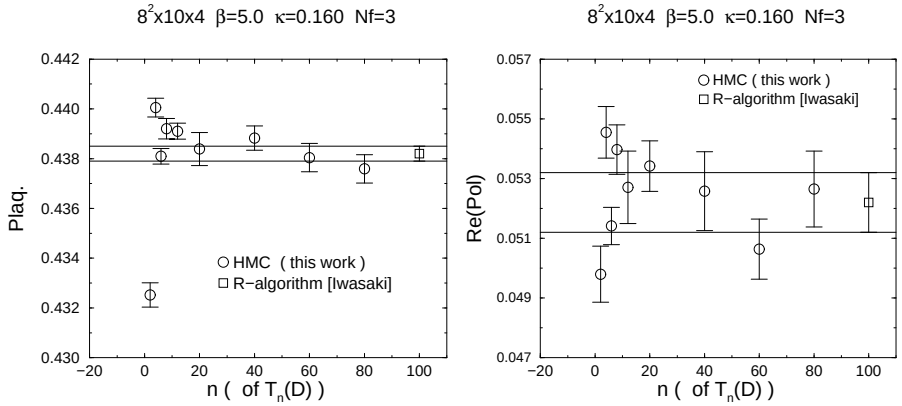

Figure 4: (left): Plaquette of $n_f = 3$ flavor QCD on an $8^2 \times 10 \times 4$ lattice at $\beta = 5.0$ and at $\kappa = 0.160$ as a function of degree n. (right): Real part of Polyakov loop.

step-size. In principle the HMC algorithm is able to simulate any flavors of QCD, with arbitrary accuracy and without extrapolation [as long as all Dirac eigenvalues are not real negative]. However the rounding errors should be under control when we use a large lattice or/and small quark masses where one may need a polynomial of high degree n to achieve sufficient approximation.

Acknowledgements

This work is partially supported by the Grant-in-Aid for Scientific Research by Monbusho, Japan(No.11740159).

References

1. S.Duane, A.Kennedy, B.Pendleton and D.Roweth, *Phys. Lett.* B **195**, 216 (1987)
2. S. Gottlieb, W.Liu, D.Toussaint, R.L.Renken and R.L.Sugar, *Phys. Rev.* D **35**, 2531 (1987).
3. For yet another approach, see T.Lippert, hep-lat/0007029, Lect.Notes Comput.Sci. and Eng. 15 (2000) 166-177.
4. M. Lüscher, *Nucl. Phys.* B **418**, 637 (1994);
   B.Bunk, K.Jansen, B.Jegerlehner, M.Lüscher, H.Simma and R.Sommer, *Nucl. Phys.* B(Proc.Suppl.) **42**, 49 (1995)
5. A. Boriçi and Ph. de Forcrand, *Nucl. Phys.* B **454**, 645 (1995).

6. C. Alexandrou, A.Boriçi, A.Feo, Ph. de Forcrand, A.Galli, F.Jegerlehner and T.Takaishi, *Phys. Rev.* D **60**, 034504 (1999).
7. Ph. de Forcrand and T. Takaishi *Nucl. Phys.* B(Proc.Suppl.) **53**, 968 (1997)
8. Y. Iwasaki, K.Kanaya, S.Kaya, S.Sakai and T.Yoshie, *Phys. Rev.* D **54**, 7010 (1996).