DESY 97-020

CERN-TH/97-21

A Polynomial Hybrid Monte Carlo Algorithm

Roberto Frezzotti
Deutsches Elektronen-Synchrotron DESY,
Notkestr. 85, D-22603 Hamburg, Germany

Karl Jansen
CERN Theory Division
CH-1211 Genève 23, Switzerland

February 14, 1997

Abstract

We present a simulation algorithm for dynamical fermions that combines the multiboson technique with the Hybrid Monte Carlo algorithm. We find that the algorithm gives a substantial gain over the standard methods in practical simulations. We point out the ability of the algorithm to treat fermion zeromodes in a clean and controllable manner.

In this letter we want to present a new algorithm for simulations of dynamical fermions. Its basic conceptual idea is to separate out the low–lying eigenvalues of the Wilson-Dirac operator on the lattice and to not take this part of the spectrum into account for the generation of the gauge field configurations. However, the algorithm can be made exact by incorporating the low–lying eigenvalues into the observables, or, alternatively, by adding them via a reject/accept step.

From a principle point of view, the separation of the eigenvalue spectrum into a high and low frequency part allows to monitor the low–lying eigenvalues. In particular, the algorithm offers the possibility to detect the appearance of eventual zeromodes and to control their effects on physical observables. The low–lying eigenvalues are also expected to play an important role in practice as they slow down the fermion simulation algorithms, when approaching the chiral limit. Cutting these modes off, should therefore result in a gain for the cost of a practical simulation.

The basic building blocks of the algorithm are the standard HMC algorithm [1] and the multiboson technique to simulate dynamical fermions [2]. A similar idea has been presented shortly in [3]. In the multiboson technique, the inverse fermion matrix is approximated by a polynomial written in powers of the fermion matrix. We propose to take this polynomial to define the –approximate– interaction of the fermions.

To be specific, let us consider the path integral for Wilson fermions on the lattice

$$\mathcal{Z} = \int \mathcal{D}U \exp \left\{-S_g\right\} \det(Q^2) = \int \mathcal{D}U\mathcal{D}\phi^\dagger\mathcal{D}\phi \exp \left\{-S_g - \phi^\dagger Q^{-2}\phi\right\} \quad . \tag{1}$$

The term S_g in the exponential is the pure gauge action and is given by

$$S_g = -\frac{\beta}{6}\sum_P Tr(U_P + U_P^\dagger) \quad . \tag{2}$$

The symbol U_P represents the usual plaquette term on the lattice with gauge links taken from SU(3). The determinant factor $\det(Q^2)$ accounts for the contribution of virtual fermion loops to the path integral. The bosonic fields ϕ carry spinor, flavour and colour indices. In eq.(1) and in the following we are assuming that we have two mass-degenerate flavours. The matrix Q that appears in the determinant is a hermitian sparse matrix defined by:

$$Q(U)_{x,y} = c_0\gamma_5[\delta_{x,y} - \kappa\sum_\mu(1-\gamma_\mu)U_{x,\mu}\delta_{x+\mu,y} + (1+\gamma_\mu)U^\dagger_{x-\mu,\mu}\delta_{x-\mu,y}] \quad , \tag{3}$$

with κ the so-called hopping parameter, related to the bare quark mass m_0 by $\kappa = (8+2m_0)^{-1}$, and $c_0 = [c_M(1+8\kappa)]^{-1}$, where c_M should be chosen such that the eigenvalues λ of Q satisfy $|\lambda| < 1$.

Let us assume that we have constructed a polynomial P_n of degree n such that

$$\det\left[Q^2 P_n(Q^2)\right] \rightarrow 1 \;\text{ for }\; n \rightarrow \infty \;, \tag{4}$$

with $P_n(\lambda(Q^2)) > 0$ for all the eigenvalues $\lambda(Q^2)$ of Q^2 in the range $0 \le \lambda(Q^2) < 1$. Then we can rewrite the determinant,

$$\det(Q^2) = \frac{\det\left[Q^2 P_n(Q^2)\right]}{\det\left[P_n(Q^2)\right]} \; . \tag{5}$$

Each of the two determinants on the right–hand side can be represented as a Gaussian integral with the help of bosonic fields ϕ and η, respectively. The partition function becomes

$$\mathcal{Z} = \int \mathcal{D}U\mathcal{D}\phi^\dagger\mathcal{D}\phi\mathcal{D}\eta^\dagger\mathcal{D}\eta W \exp\left\{-S_g - \phi^\dagger P_n(Q^2)\phi - \eta^\dagger\eta\right\} \tag{6}$$

where we introduced the "correction factor"

$$W = \exp\left\{\eta^\dagger\left(1 - \left[Q^2 P_n(Q^2)\right]^{-1}\right)\eta\right\} \; . \tag{7}$$

Note that eq.(6) is an exact rewriting of the partition function eq.(1).

With the introduction of the correction factor W the expectation value of an observable O is now computed as

$$< O >= \frac{< OW >_P}{< W >_P} \; , \tag{8}$$

where the averages $< \ldots >_P$ are taken with respect to the measure defined through the approximate fermion action $\phi^\dagger P_n(Q^2)\phi$. Alternatively one may incorporate the W factor via a reject/accept step.

Let us now specify the form of the polynomial that we are going to take. We choose a Chebyshev polynomial to approximate Q^{-2}. When written in its factorized form

$$P_{[n,\epsilon]}(Q^2) = c_N \prod_{k=1}^{n}\left[Q^2 - z_k\right] = c_N \prod_{k=1}^{n}\left[(Q - \sqrt{z_k}^*)(Q - \sqrt{z_k})\right] \; , \tag{9}$$

it is characterized by its roots (for $k = 1, 2, \ldots, n$),

$$z_k = \frac{1}{2}(1+\epsilon) - \frac{1}{2}(1+\epsilon)\cos\left(\frac{2\pi k}{n+1}\right) - i\sqrt{\epsilon}\sin\left(\frac{2\pi k}{n+1}\right) \tag{10}$$

and a normalization factor c_N, which is explicitly calculable [7]. The polynomial $P_{[n,\epsilon]}(s)$ approximates the function $1/s$ (where s may correspond to any of the eigenvalues of Q^2) uniformly in the interval $\epsilon \le s \le 1$. The relative fit error

$$R_{[n,\epsilon]}(s) = \left[P_{[n,\epsilon]}(s) - 1/s\right] s \tag{11}$$

in this interval is exponentially small:

$$|R_{[n,\epsilon]}(s)| \le 2\left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)^{n+1} \; . \tag{12}$$

Let us finally introduce an accuracy parameter δ, which is actually an upper bound to the maximum relative error of the polynomial approximation,

$$\delta = 2\left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)^{n+1} \; . \tag{13}$$

The parameter δ provides an easily computable and conservative measure of how well the chosen polynomial approximates $1/s$ in the given interval $\epsilon \le s \le 1$.

With the specification of the polynomial eq.(9) the path integral eq.(6) and the correction factor W are fully determined. It is clear that for polynomials of high degree, the interaction defined by them becomes too complicated for the application of local algorithms. It is therefore a natural choice to use molecular dynamics algorithms like the HMC or the Kramers equation algorithms [4, 5]. In the following we will call our hybrid of molecular dynamics and multiboson algorithms the Polynomial Hybrid Monte Carlo (PHMC) algorithm.

What do we expect from the PHMC algorithm? It has been suggested [6] that the lowest eigenvalue of Q^2, $\lambda_{min}(Q^2)$, is an important quantity in determining the cost of the HMC algorithm. In particular, a theoretical analysis leads to the –optimistic– estimate that the cost grows as $1/\lambda_{min}^{3/2}(Q^2)$. In the PHMC algorithm, the role of the lowest eigenvalue is taken over by the infrared cut-off parameter ϵ. For $0 \le s \le \epsilon$ the polynomial $P_{[n,\epsilon]}(s)$ is always finite with values $O(1/\epsilon)$. From the experience with the multiboson technique [7, 8, 9, 10, 11] it has become clear that one might choose ϵ to be substantially larger than $\lambda_{min}(Q^2)$ while still getting values for expectation values that are compatible with the ones obtained by the HMC algorithm. This result suggests that one might choose $\epsilon > \lambda_{min}(Q^2)$ also in the PHMC algorithm without introducing too large fluctuations of the correction factor. Since in the PHMC algorithm only one bosonic field is introduced, one will also avoid the dangerous increase of the autocorrelation time with increasing degree of the polynomial as found for the multiboson technique [8].

Before we turn to the results for the performance of the PHMC algorithm, let us shortly sketch, how the algorithm is implemented in our simulation program. We will be quite short here und refer to a forthcoming publication for more details and safety measures, in particular when using the algorithm on a 32-bit arithmetics machine. Let us start by discussing the heatbath for the bosonic fields ϕ, the action of which is given by

$$S_b = \phi^\dagger P_{[n,\epsilon]}(Q^2)\phi \, . \tag{14}$$

To generate a Gaussian distribution according to this interaction, we proceed as follows. We first generate a Gaussian random vector ζ. We then solve $Q^2 P_{[n,\epsilon]}(Q^2)X = Q^2\zeta$ using a Conjugate Gradient (CG) method. By writing $P_{[n,\epsilon]}(Q^2) = P^*_{n/2}(Q^2)P_{n/2}(Q^2)$ with appropriate ordering of the roots, we finally construct the ϕ-fields via $\phi = P^*_{n/2}(Q^2)X$.

The derivation of the force for the PHMC algorithm is done in complete analogy to the method used for the HMC algorithm [12]. A variation of the action eq.(14) using the polynomial eq.(9) reveals that one has to construct the vectors

$$\phi_j = \prod_{k=1}^{j} [Q - \sqrt{z_k}]\, \phi_0 \;\; ; j = 1, ..., 2n-1 \tag{15}$$

with ϕ_0 the bosonic field generated by the boson heat bath. The vectors ϕ_j are precalculated and stored. This calculation may be organized in such a way that the memory storage required amounts to only $n+1$ (instead of $2n$) vectors ϕ_j. One may then use them in the actual force computation at the appropriate places.

The above storage requirement for the vectors ϕ_j may be further reduced by introducing more bosonic field copies. For example, one may split the polynomial into two parts $P^{(1)}_{n/2}$ and $P^{(2)}_{n/2}$ satisfying $det\left[P_{[n,\epsilon]}(Q^2)\right] = det\left[P^{(1)}_{n/2}(Q^2)\right]\cdot det\left[P^{(2)}_{n/2}(Q^2)\right]$ and integrate each contribution separately, leading to the action

$$S_b = \phi_1^\dagger P^{(1)}_{n/2}(Q^2)\phi_1 + \phi_2^\dagger P^{(2)}_{n/2}(Q^2)\phi_2 \; . \tag{16}$$

It is amusing to note that by iterating this procedure one can obtain an interpolation between a nearly exact HMC algorithm, if n and $1/\epsilon$ are large enough, and the multiboson technique to simulate dynamical fermions. Although we did not yet perform an extended analysis, our first results indicate that introducing a few bosonic field copies does not increase the autocorrelation time, when the number of copies is held small, say less than about eight. It remains a subject of further study, however, whether the molecular dynamics behaviour is severely altered by the introduction of more field copies.

Finally, the computation of the correction factor W eq.(7) needs an additional inversion of $Q^2P_{[n,\epsilon]}(Q^2)$. Since the η-field occurring in W is completely independent from the ϕ-field in the boson heatbath, this inversion has to be done separately.

We decided to test the PHMC against the HMC algorithm on 4^4 and 8^4 lattices. All numerical results have been obtained on Alenia Quadrics (APE) massively parallel computers. We adopted Schrödinger functional boundary conditions. For the 4^4 lattice we ran at $\beta = 6.4$, $\kappa = 0.15$ and for the 8^4 lattice we had $\beta = 5.6$ and $\kappa = 0.1585 \approx \kappa_c$ [13]. In both, the HMC and the PHMC algorithms, even-odd preconditioning [14] and a Sexton-Weingarten leap-frog integration scheme [15] is implemented. We want to emphasize that most of the improvements to accelerate the HMC algorithm can be taken over to the PHMC algorithm. We think, therefore, that the results of the comparison we are performing here should be independent of the particular implementation.

Table 1: Technical parameters for both algorithms

	HMC		PHMC				
Lattice	ϵ_{md}	N_{md}	ϵ_{md}	N_{md}	ϵ	n	c_M
4^4	0.25	4	0.25	4	0.036	12	0.5789
8^4	0.075	13	0.09	10	0.0026	48	0.5789

In table 1, we give the parameters of the algorithms which are the step size ϵ_{md} and the number of molecular dynamics steps N_{md} as used for the leap frog integration. We also give the parameters characterizing the polynomial. The parameters were tuned in such a way that about the same acceptance rate was achieved in both algorithms, namely 82% and 86% for the

HMC and PHMC algorithms, respectively, on the 4^4 and, correspondingly, 80% and 79% for the 8^4 lattices.

With the choice of c_M in table 1 we found the value of the largest eigenvalue $\lambda_{max}(\hat{Q}^2)$ of the preconditioned matrix $\hat{Q}$ to be close to 1. As first noted in [19], since the polynomial is constructed such that it provides a uniform approximation of $\hat{Q}^{-2}$ for $\epsilon < \lambda < 1$, lifting the eigenvalues by choosing $c_M < 1$, allows to choose a larger value of ϵ and therefore a polynomial of lower degree in order to achieve a desired value for the accuracy parameter δ.

We give in table 2 results for the expectation values of the plaquette $< P >$ and the lowest eigenvalue $< \lambda_{min}(\hat{Q}^2) >$. We also give the uncorrected expectation values (setting $W = 1$ in eq.(7)) denoted by a $*$. In the third column we give the number of trajectories that were taken for the analysis.

Table 2: Results for both algorithms

Lattice	Algorithm	# trajectories	$\langle P \rangle$	$\langle \lambda_{min}(\hat{Q}^2) \rangle$
4^4	HMC	18000	0.66179(13)	0.01582(9)
	PHMC	18000	0.66169(16)	0.01570(10)
	PHMC*	18000	0.66248(13)	0.01324(7)
8^4	HMC	2745	0.57251(12)	0.001310(51)
	PHMC	2560	0.57253(16)	0.001328(51)
	PHMC*	2560	0.57272(14)	0.001141(51)

First of all, table 2 confirms the correctness of the PHMC algorithm. While on the 4^4 lattice the correction factor is important, one notices that for the 8^4 lattice it only has a small effect. A crucial question is, whether the correction factor introduces strong fluctuations that may lead to large errors for the corrected observables. We find that this is not the case, when we arrange for a situation where ϵ is 2–3 times larger than the lowest eigenvalue of the problem and the relative fit error of the polynomial is kept small enough, $\delta \lesssim 0.02$. For larger values of δ the fluctuations can become substantial, leading to large errors for the corrected observables eq.(8).

In addition, the fluctuations of the corrected observables can be suppressed further by choosing the number of updates of the η-fields to be larger than the number of full gauge field updates. This amounts to compute the correction factor N_{corr} times on the same gauge field configuration. In our test, presented here, we have chosen $N_{corr} = 1$ on the 4^4 lattice and $N_{corr} = 2$ on the 8^4 lattice.

Since the behaviour of the observables from the HMC and the corrected ones eq.(8) from the PHMC algorithm may in principle be very different, it is important to find an estimate for the true error in order to be able to compare both algorithms. To this end, we used a jack-knife binning procedure, looking for a plateau in the blocked errors. For the HMC algorithm, as a consistency check, we determined also the integrated autocorrelation time computed directly

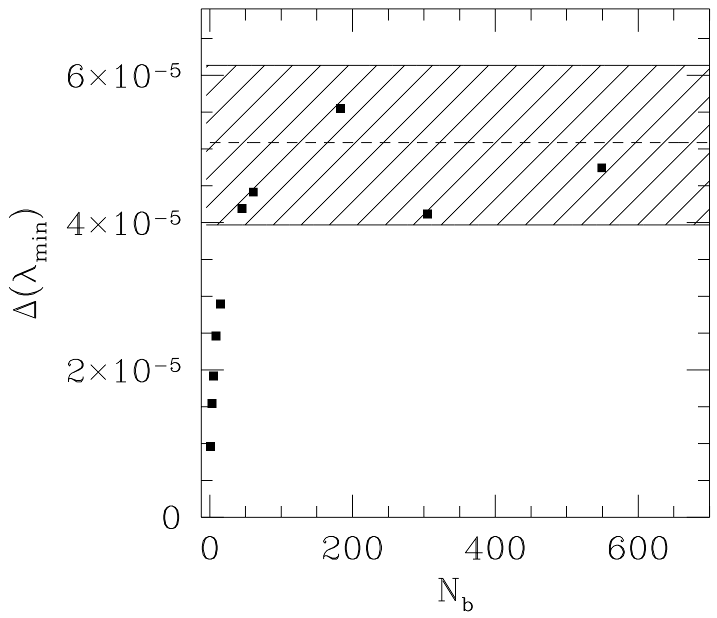

Figure 1: The blocked jack-knife errors for the lowest eigenvalue $\lambda_{min}(\hat{Q}^2)$ from the HMC algorithm (filled squares). N_b is the binning block length. The dashed line ist the estimate for the blocked error from the integrated autocorrelation time in the HMC algorithm. The shaded region is the estimate for the true error from the PHMC algorithm.

from the autocorrelation function. The result is illustrated in fig. 1 for the case of the lowest eigenvalue of $\hat{Q}^2$ on the 8^4 lattice. The filled squares are the blocked errors $\Delta(\lambda_{min}(\hat{Q}^2))$ as a function of the block length N_b as obtained from the HMC algorithm. $N_b = 1$ corresponds to the naive error Δ_{naive}, $N_b = 2$ to blocking two consecutive measurements and so on. The dashed line indicates the true value of the error as computed from the integrated autocorrelation time τ, $\Delta_{true} = \sqrt{2\tau}\Delta_{naive}$.

For the PHMC algorithm on the 8^4 lattice we ran on the QH2 version of the APE machine with 256 nodes. Distributing the 8^4 lattice on 8 of these nodes, gives us 32 independent systems, from which the error can be evaluated straightforwardly. One may also build from these 32 systems 2 groups, each consisting of 16 independent systems and giving a separate error estimate Δ_1 and Δ_2. We take the difference between Δ_1 and Δ_2 as an estimate of the "error of the error". We plot this uncertainty of the error as the shaded region in fig. 1. The same analysis can be made for the plaquette with a similar result.

We conclude that with the same number of trajectories both algorithms give compatible error bars for the plaquette and the lowest eigenvalue. Note, however, that with our statistics the error on the error is still significant. This is, of course, just a reflection of the uncertainty in the determination of the autocorrelation times.

Let us discuss now the cost of a single trajectory in both algorithms. We write the total cost for the algorithms as

$$C_{tot} = C_{Q\phi} + C_{extra} \ , \tag{17}$$

where the first contribution is given by the number of matrix times vector $Q\phi$ operations and the second part accounts for all other operations. Asymptotically, when the condition number of Q becomes large, $C_{Q\phi}$ will by far dominate the cost of the algorithms. We will therefore only discuss and compare the cost $C_{Q\phi}$ in the following. Let us remark, however, that for small condition numbers C_{extra} can be a non-negligible part of the total cost, in particular for the HMC algorithm as one might deduce from the details of the algorithm structure.

Let us denote by N_{CG} the average number of iterations of the Conjugate Gradient algorithm that is implemented in our programs for all matrix inversions. Then the cost for the HMC algorithm in units of $Q\phi$ operations is given by

$$C_{Q\phi}(HMC) = 2 \cdot (2N_{md} + 1) \cdot N_{CG} \ , \tag{18}$$

where the first factor of 2 stems from the fact that one needs 2 $Q\phi$ operations in each iteration of the CG routine. The factor $(2N_{md} + 1)$ originates from the use of the Sexton-Weingarten integration scheme [15]. The cost for the PHMC algorithm is split into three parts,

$$C_{Q\phi}(PHMC) = C_{bhb} + C_{update} + C_{corr} \ , \tag{19}$$

where C_{bhb} is the cost for the heatbath of the bosonic fields, C_{update} the cost for the force computation and C_{corr} the cost to evaluate the correction factor. In units of $Q\phi$ operations we find

$$\begin{aligned} C_{bhb} &= 2n \cdot N_{CG}^{bhb} + n \\ C_{update} &= 3n \cdot (2N_{md} + 1) \\ C_{corr} &= 2n \cdot N_{CG}^{corr} \cdot N_{corr} \ . \end{aligned} \tag{20}$$

The factor N_{corr} denotes as above the number of evaluations of the correction factor W per full gauge field update. The factor $3n$ in C_{update} comes for the following reason. One needs basically $2n$ $Q\phi$ operations to construct the fields ϕ_j of eq.(15). The computation of the total force needs a loop over the number of fields, n. In each iteration of this loop one has to compute the variation of the action with respect to the gauge fields for all four directions. This computation corresponds roughly to one $Q\phi$ multiplication. We explicitly verified this expectation for our implementation of the PHMC algorithm on the APE computer. We expect, however, the formula for C_{update} in the PHMC algorithm to also hold for other situations like improved Wilson fermions.

We give the cost of both algorithms in table 3. We see that on the 8^4 lattice, we win about a factor of 1.8 against the HMC algorithm. Let us make a few remarks at this point.

Table 3: Cost for both algorithms

Lattice	Algorithm	C_{bhb}	C_{update}	C_{corr}	$C_{Q\phi}$
4^4	PHMC	130	324	86	540
	HMC	—	868	—	868
8^4	PHM	350	3024	600	3974
	HMC	—	7398	—	7398

(Correction factor) We find for our PHMC algorithm that the plaquette has an autocorrelation time of two while for the lowest eigenvalue the autocorrelation time comes out to be about four. In this situation it is not necessary to calculate the correction factor for every trajectory. Indeed, the correction factor should be more considered as part of the measurement and, from this point of view, its cost should be added to the measurement costs. For observables, for which the measurements are time consuming, or for situations where not every trajectory is measured, the cost of the correction factor becomes negligible.

(Long trajectories) If we go to larger lattices and situations with larger condition numbers than considered here, the number of steps per trajectory, N_{md}, is increased when the trajectory length is kept fixed. We therefore expect that again the overheads for the correction factor and also for the boson heatbath will become negligible. From the above numbers, we conclude that the cost for the update itself, C_{update}, is reduced in the PHMC algorithm by more than a factor of two as compared to standard HMC.

(Parallelization) When using massively parallel architectures, it is often advantageous to simulate several lattices simultaneously. If one uses the HMC algorithm, on SIMD architectures all the systems have to wait until the system with the largest number of CG iterations has converged. If one compares the number of CG iterations from running 32 replicas in parallel, $N_{CG}(32 \text{ systems})$, to the one from running only a single system, $N_{CG}(1 \text{ system})$, one finds that the ratio $N_{CG}(32 \text{ systems})/N_{CG}(1 \text{ system})$ may easily reach values of about 2. In the PHMC algorithm, at least in the most time consuming part, the number of $Q\phi$ operations is, however, fixed by the degree of the polynomial and the same for each system. We expect therefore for this situation the PHMC algorithm to give an additional gain. Let us emphasize that, of course, all of the costs of the algorithms given in table 3 refer to the case of running a single system.

In conclusion, we have presented a new algorithm, called the PHMC algorithm which is a hybrid of the standard HMC algorithm and the multiboson technique to simulate dynamical fermions. Within the uncertainty of the error determination, shown in fig. 1, we find that for the same number of trajectories, the errors from the HMC and the PHMC algorithms are about equal. At the same time, the cost of generating a single trajectory is reduced by almost a factor of 2 when using the PHMC algorithm. Certainly, the properties of the PHMC algorithm have to be investigated more, different observables should be considered and, of course, improved fermions should be studied. However, we find our results very promising to finally find a real gain of about a factor of 2 over the standard HMC algorithm.

As a rule of thumb we advise to choose the lowest end of the fit range, ϵ, two or three times larger than the lowest eigenvalue of $\hat{Q}^2$ and the degree n of the fitting polynomial such that the accuracy parameter δ, introduced in eq.(13), is between 0.01 and 0.02. For too large values of δ, the fluctuations of the corrected observables eq.(8) become too large. For too small values, the degree of the polynomial increases too much.

Even more than the practical gain that we anticipate, we think that the PHMC algorithm has an advantage which is of principle nature. It has been demonstrated that for Wilson fermions, with and without Symanzik improvement, fermionic (almost) zeromodes may appear in the quenched approximation [16, 17]. Such modes distort the statistical sample substantially. On the other hand, as discussed in [17], the full path integral is finite and the fermion zeromodes are cancelled by the measure.

The way the standard fermion simulation algorithms deal with the zeromodes, leaves us with a dilemma. Either these algorithms suppress the zeromodes so strongly that in practical simulations configurations carrying (almost) zeromodes do not occur at all. But then we do not know what their importance is on physical observables, which is unfortunate in particular within the context of topology. Or, on the other hand, a few configurations with (almost) fermion zeromodes are actually generated. But then they will lead to exceptional values for quark propagators and a reliable measurement of the observables involving them will become very difficult.

In our PHMC algorithm, the update part is safe against the zeromodes, since the infrared cut-off parameter ϵ leaves the polynomial always finite. One may, however, monitor the lowest eigenvalue and its eigenvector during a simulation by using minimization techniques like the one described in [18]. If an *isolated* zeromode is detected, one may switch from the computation of the correction factor discussed above to the following strategy. What we want to compute is

$$det\left[Q^2 P(Q^2)\right] \equiv det\,[A]\ . \tag{21}$$

Since we know the lowest eigenvalue $\lambda_{min}(A) = \lambda_{min}(Q^2)P(\lambda_{min}(Q^2))$ and its eigenvector χ, we may define a projector P_χ that projects onto the subspace orthogonal to χ leading to a matrix, where the lowest eigenvalue is taken out,

$$\tilde{A} = A - \lambda_{min}(A)P_\chi\ . \tag{22}$$

Now, it is not difficult to show that

$$det\,[A] = \lambda_{min}(A)det\left[\tilde{A}\right]\ , \tag{23}$$

where the factor $det\left[\tilde{A}\right]$ may again be evaluated with the help of gaussian bosonic fields $\tilde{\eta}$.

For pure gauge observables, like Wilson loops, we find therefore that the configurations carrying zeromodes have a negligible weight in eq.(8). For a fermionic observable involving quark propagators, the situation is different because in the numerator of eq.(8) the zeromode configurations may give a finite, non–vanishing contribution, while in the denominator these configurations do not contribute. In this case the strategy will be to again separate out the leading divergent contribution to the observable, which, when considering two degenerate quark flavours, may be at most proportional to $\lambda_{min}^{-1}(Q^2)$. Technically, this can be achieved again by projecting the lowest eigenvalue out. The divergence possibly appearing in the leading contribution will now be cancelled by the infinitesimal factor (proportional to $\lambda_{min}(Q^2)$) from the correction factor yielding, as expected, a finite, non–vanishing, well defined result. The non–leading contributions to the fermionic observable (*i.e.* the ones less divergent than $\lambda_{min}^{-1}(Q^2)$) may also be evaluated, by basically inverting the matrix $Q^2 - \lambda_{min}(Q^2)P_\chi$, which is now well conditioned. Note that these contributions are suppressed by the correction factor as $\lambda_{min}(Q^2) \to 0$.

The above discussion may be generalized to a situation where a number of eigenvalues assume very small values. We therefore find that our PHMC algorithm is in principle able to take eventual zeromodes into account in a controllable way when performing dynamical fermion simulations.

Acknowledgements

This work is part of the ALPHA collaboration research program. We thank M. Lüscher for essential comments and very helpful discussions, in particular with respect to the role of zeromodes in fermion simulation algorithms. All numerical simulation results have been obtained on the Alenia Quadrics (APE) computers at DESY-IFH (Zeuthen). We thank the staff of the computer center at Zeuthen for their support. R.F. thanks the Alexander von Humboldt Foundation for the financial support to his research stay at DESY.

References

[1] S. Duane, A. D. Kennedy, B. J. Pendleton and D. Roweth, Phys. Lett. B195 (1987) 216.

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

[3] P. de Forcrand and T. Takaishi, hep-lat/9608093.

[4] A. M. Horowitz, Phys. Lett. 156B (1985) 89; Nucl. Phys. B280 (1987) 510; Phys. Lett. 268B (1991) 247.

[5] K. Jansen and C. Liu, Nucl. Phys. B453 (1995) 375; ibid. B459 (1996) 437.

[6] K. Jansen, review talk at the International Symposium on Lattice Field Theory, 1996, St. Louis, Mo, USA, hep-lat/9607051.

[7] B. Bunk, K. Jansen, B. Jegerlehner, M. Lüscher, H. Simma and R. Sommer, hep-lat/9411016; Nucl. Phys. B (Proc. Suppl.) 42 (1995) 49.

[8] B. Jegerlehner, Nucl.Phys.B465 (1996) 487.

[9] C. Alexandrou, A. Borrelli, P. de Forcrand, A. Galli and F. Jegerlehner, Nucl.Phys.B456 (1995) 296.

[10] A. Borrelli, P. de Forcrand and A. Galli, Nucl.Phys. B477 (1996) 809.

[11] K. Jansen, B. Jegerlehner and C. Liu, Phys.Lett. B375 (1996) 255.

[12] S. Gottlieb, W. Liu, D. Toussaint, R. L. Renken and R. L. Sugar, Phys. Rev. D 35 (1987) 2531.

[13] R. Gupta et.al., Phys.Rev.D40 (1989) 2072.

[14] T. Degrand and P. Rossi, Comp. Phys. Comm. 60 (1990) 211.

[15] J. C. Sexton and D. H. Weingarten, Nucl. Phys. B380 (1992) 665.

[16] K. Jansen, C. Liu, H. Simma and D. Smith, hep-lat/9608048.

[17] M. Lüscher, S.Sint, R. Sommer, P. Weisz and U. Wolff, CERN preprint, CERN-TH/96-218, hep-lat/9609035.

[18] T, Kalkreuter and H. Simma, Comp.Phys.Comm. 93 (1996) 33.

[19] A. Borici and P. de Forcrand, Nucl.Phys.B454 (1995) 645.