CERN/TH-99-376
CPT-99/PE.3916

LAPTH-Conf-771/99

A numerical treatment of Neuberger's lattice Dirac operator*

Pilar Hernández[†], Karl Jansen[‡]

CERN, 1211 Geneva 23, Switzerland

and Laurent Lellouch[§]

LAPTH, Chemin de Bellevue, B.P. 110,
F74941 Annecy-le-Vieux Cedex, France

January 13, 2000

Abstract

We describe in some detail our numerical treatment of Neuberger's lattice Dirac operator as implemented in a practical application. We discuss the improvements we have found to accelerate the numerical computations and give an estimate of the expense when using this operator in practice.

[*] Talk given by K.J. at the "Interdisciplinary Workshop on Numerical Challenges to Lattice QCD", Wuppertal, August 22-24,1999

[†] On leave from Departamento de Física Teórica, Universidad de Valencia.

[‡] Heisenberg Foundation Fellow

[§] On leave from Centre de Physique Théorique, CNRS Luminy, F-13288 Marseille Cedex 9, France.

1 Lattice formulation of QCD

Today, we believe that the world of quarks and gluons is described theoretically by quantum chromodynamics (QCD). This model shows a number of non-perturbative aspects that cannot be adequately addressed by approximation schemes such as perturbation theory. The only way to evaluate QCD, addressing both its perturbative and non-perturbative aspects *at the same time*, is lattice QCD. In this approach the theory is put on a 4-dimensional Euclidean space-time lattice of finite physical length L, with a non-vanishing value of the lattice spacing a. Having only a finite number of grid points, physical quantities can be computed numerically by solving a high-dimensional integral by Monte Carlo methods, making use of importance sampling.

The introduction of a lattice spacing regularizes the theory and is an intermediate step in the computation of physical observables. Eventually, the regularization has to be removed and the value of the lattice spacing has to be sent to zero to reach the target theory, i.e. continuum QCD. In fact, in conventional formulations of lattice QCD [1], the introduction of the lattice spacing renders the theory on the lattice somewhat different from the continuum analogue and a number of properties of the continuum theory are only very difficult and cumbersome to establish in the lattice regularized theory. One of the main reasons for this difficulty is that in conventional lattice QCD the regularization breaks a particular symmetry of the continuum theory, which plays a most important role there, namely chiral symmetry.

However, the last few years have seen a major breakthrough in that we now have formulations of lattice QCD that have an exact lattice chiral symmetry [5]. In this approach, many properties of continuum QCD are preserved *even at a non-vanishing value of the lattice spacing* [2, 3, 4, 5, 6]. This development followed the rediscovery [7] of the so-called Ginsparg–Wilson (GW) relation [8] which is fulfilled by any operator with the exact lattice chiral symmetry of [5]. It is not the aim of this contribution to discuss the physics consequences of the GW relation. We have to refer the interested reader to reviews [9, 10] about these topics. Here we would like to discuss the *numerical* treatment of a particular lattice operator that satisfies the GW relation, namely Neuberger's solution [4]. This solution has a complicated structure and is challenging to implement numerically. Thus, the large theoretical advantage of an operator satisfying the GW relation must be weighed against the very demanding computational effort required to implement it.

This contribution is organized as follows. After discussing Neuberger's lattice Dirac operator we want to show how we evaluated the operator in our practical application [11] and what kind of improvements we found to accelerate the numerical computations. For alternative ideas for improvements, see the contributions of H. Neuberger [12] and A. Borici [13] to this workshop. We finally give some estimates of the computational expense of using Neuberger's operator.

2 Neuberger's lattice Dirac operator

The operator we have used acts on fields (complex vectors) $\Phi(x)$ where $x = (x_0, x_1, x_2, x_3)$ and the x_μ, $\mu = 0, 1, 2, 3$, are integer numbers denoting a 4-dimensional grid point in a lattice of size N^4 with $N = L/a$. The fields $\Phi(x)$ carry in addition a "colour" index $\alpha = 1, 2, 3$ as well as a "Dirac" index $i = 1, 2, 3, 4$. Hence, Φ is a $N^4 \cdot 3 \cdot 4$ complex vector.

In order to reach the expression for Neuberger's operator we first introduce the matrix A

$$A = 1 + s - \frac{a}{2}\left\{\gamma_\mu\left(\nabla^*_\mu + \nabla_\mu\right) - a\nabla^*_\mu\nabla_\mu\right\} \ , \tag{1}$$

where ∇_μ and ∇^*_μ are the nearest-neighbour forward and backward derivatives, the precise definition of which can be found in the Appendix. The parameter s is to be taken in the range of $|s| < 1$ and serves to optimize the localization properties [14] of Neuberger's operator, which is then given by

$$D = \frac{1}{a}\left\{1 - A\left(A^\dagger A\right)^{-1/2}\right\}. \tag{2}$$

Through the appearance of the square root in eq. (2), all points on the lattice are connected with one another, giving rise to a very complicated, multi-neighbour action. However, the application of D to a vector Φ will only contain applications of A or $A^\dagger A$ on this vector. Since these matrices are sparse, as only nearest-neighbour interactions are involved, we will never have to store the whole matrix. In the computation of physical quantities, the inverse of D, applied to a given vector, is generically needed. Hence one faces the problem of having to compute a vector $X = D^{-1}\eta$, with η a prescribed vector (the "source") as required by the particular problem under investigation. Fortunately, a number of efficiently working algorithms for computing $X = D^{-1}\eta$ are known, such as conjugate gradient, BiCGstab, or variants thereof [16]. In conventional approaches to lattice QCD an operator $\tilde{D}$ is used that is very similar to the matrix A in eq. (1). Computing the vector $\tilde{X} = \tilde{D}^{-1}\eta$ requires a number n_{iter} of iterations of some particular method, say BiCGstab. Employing Neuberger's operator D in computing $X = D^{-1}\eta$, it turns out that the number of iterations needed is of the same order of magnitude as when using $\tilde{D}$. At the same time, in each of these iterations, the square root has to be evaluated. When this is done by some polynomial approximation, it is found that the required degree of this polynomial is roughly of the same order as the number of iterations needed for computing the vector X. Hence, with respect to the conventional case, the numerical effort is squared and the price to pay for using the operator D is high.

On the other hand, any solution of the Ginsparg–Wilson relation gives us a tool by which particular problems in lattice QCD can be studied, which would be extremely hard to address with conventional approaches. It is for these cases that the large numerical effort is justified, but clearly, we would like to have clever ideas coming from areas such as Applied Mathematics, to decrease the numerical expense or even overcome this bottleneck.

3 Approximation of $\left(A^{\dagger}A\right)^{-1/2}$

For computing the square root that appears in eq. (2), we have chosen a Chebyshev approximation [15] by constructing a polynomial $P_{n,\epsilon}(x)$ of degree n, which has an exponential convergence rate in the interval $x \in [\epsilon, 1]$. Outside this interval, convergence is still found but it will not be exponential. The advantages of using this Chebyshev approximation are the well-controlled exponential fit accuracy as well as the possibility of having numerically very stable recursion relations [17] to construct the polynomial, allowing for large degrees. In order to have an optimal approximation, it is desirable to know the lowest and the highest eigenvalue of $A^{\dagger}A$. A typical example of the eigenvalues of $A^{\dagger}A$ is shown in fig. 1, where we show the 11 lowest eigenvalues as obtained on a number of configu-

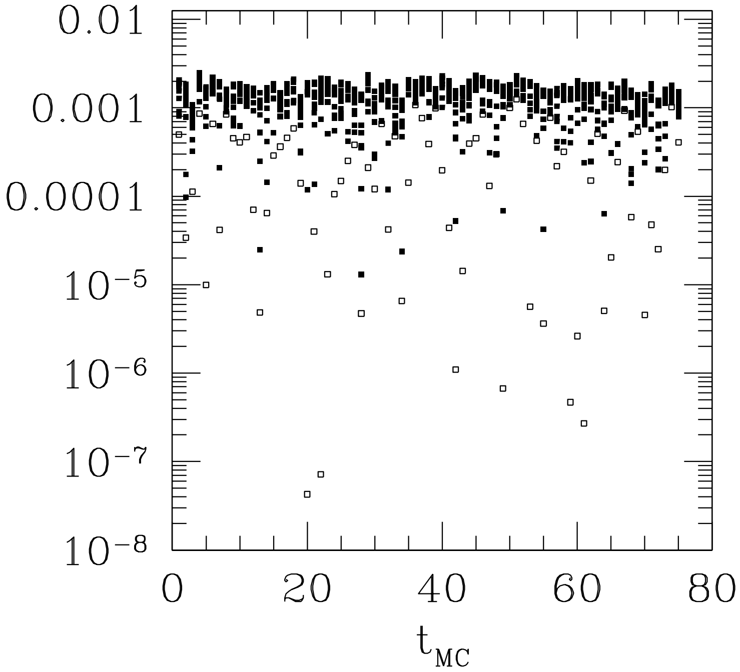

Figure 1: Monte Carlo time evolution of the eleven lowest eigenvalues of $A^\dagger A$ at $\beta = 5.85$. The lowest eigenvalue for each configuration is the open square.

rations using the Ritz functional method [19]. There is a wide spread and very low-lying eigenvalues appear. Choosing ϵ to be the value of the lowest of these eigenvalues would result in a huge degree n of the polynomial $P_{n,\epsilon}$. We therefore computed O(10) lowest-lying eigenvalues of $A^\dagger A$ as well as their eigenfunctions and projected them out of the matrix $A^\dagger A$. The approximation is then only performed for the matrix with a reduced condition number, resulting in a substantial decrease of the degree of the polynomial. In addition, we computed the highest eigenvalue of $A^\dagger A$ and normalized the matrix A such that $\|A^\dagger A\| \lesssim 1$.

Since our work [11], aiming at the physical question of spontaneous chiral symmetry breaking in lattice QCD, has been one of the first of its kind, we wanted to exclude possible systematic errors and demanded a very high precision for the approximation to the square root:

$$\|X - P_{n,\epsilon}(A^{\dagger}A)A^{\dagger}AP_{n,\epsilon}(A^{\dagger}A)X\|^2/\|2X\|^2 < 10^{-16} \tag{3}$$

where X is a gaussian random vector. In our practical applications we fixed this precision beforehand and set ϵ to be the 11th lowest eigenvalue of $A^{\dagger}A$. This then determines the degree of the polynomial n and hence our approximation D_n to the exact Neuberger operator D. We checked that the precision we required for the approximation of the square root is directly related to the precision by which the GW relation itself is fulfilled. Choosing n such that the accuracy in eq. (4) is reached results in

$$\| \left[\gamma_5 D_n + D_n\gamma_5 - D_n\gamma_5 D_n\right] X\|^2/\|X\|^2 \approx 10^{-16}\ . \tag{4}$$

In addition, we find that the deviations from the exact GW relation decrease exponentially fast with increasing n.

4 The inverse of Neuberger's operator

As mentioned above, in physics applications a vector $D^{-1}\eta$ has to be computed, with η a prescribed source vector. Not only is the computation of this vector very costly, there also appears to be a conceptual problem: in inspecting the lowest eigenvalue of $D_n^{\dagger}D_n$, very small eigenvalues are often found as shown in fig. 2. These very small eigenvalues belong to a given chiral sector of the theory, i.e. their corresponding eigenfunctions χ are eigenfunctions of γ_5 with $\gamma_5\chi = \pm\chi$. In fact, these modes play an important physical role as they are associated with topological sectors of the theory [3, 2, 5, 4].

As far as the practical applications are concerned, it is clear that in the presence of such a small eigenvalue, the inversion of D_n will be very costly, as the condition number of the problem is then very high. In order to address this problem, we followed two strategies:

(i) We compute the lowest eigenvalue of $D_n^{\dagger}D_n$ and its eigenfunction (using again the Ritz functional method [19]) and if it is a zero mode –in which

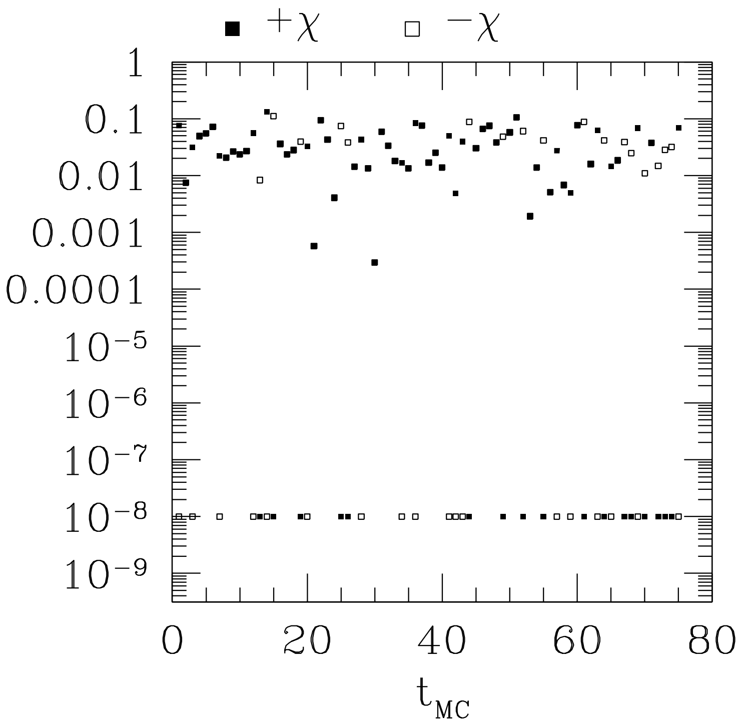

Figure 2: Monte Carlo time evolution of the lowest eigenvalue of $D_n^\dagger D_n$. The eigenvalues belong to given chiral sectors of the theory denoted as $\pm\chi$ for chirality plus (full squares) and minus (open squares). Data are obtained at $\beta = 5.85$ choosing $s = 0.6$. Whenever there is a zero mode of $D_n^\dagger D_n$, the value of the lowest eigenvalue is set to 10^{-8}.

case it is also a zero mode of D_n– we project this mode out of D_n and invert only the reduced matrix; this is then well conditioned, as the very small eigenvalues appear to be isolated. In this strategy, the knowledge of the eigenfunction must be very precise and an accuracy of approximating the square root as indicated in eq. (3) is mandatory.

(*ii*) Again we determine the lowest eigenvalue of $D_n^\dagger D_n$ and the chirality of the corresponding zero mode, if there is any. We then make use of the fact that $D_n^\dagger D_n$ commutes with γ_5. This allows us to perform the inversion in the chiral sector *without zero modes*. In this strategy, the accuracy demanded in eq. (3) could be relaxed and this strategy, which essentially follows ref. [18], is in general much less expensive than following strategy (*i*).

However, even adopting strategy (*ii*), solving the system $D_n X = \eta$ is still costly. We therefore tried two ways of improving on this. We first note that instead of solving

$$\left[1 - A/\sqrt{A^\dagger A}\right] X = \eta \tag{5}$$

we can equally well solve

$$\left[A^\dagger - \sqrt{A^\dagger A}\right] X = A^\dagger \eta \ . \tag{6}$$

In practice, however, we found no real advantage in using the formulation of eq. (6). We have further considered two acceleration schemes.

Scheme (a)

We choose two different polynomials (now approximating $\sqrt{A^\dagger A}$ and not the inverse) $P_{n,\epsilon}$ and $P_{m,\epsilon}$, $m < n$, such that

$$P_{n,\epsilon} = P_{m,\epsilon} + \Delta \tag{7}$$

with Δ a "small" correction. Then we have

$$\begin{aligned}\left[A^\dagger - P_{n,\epsilon}\right]^{-1} &= \left[A^\dagger - P_{m,\epsilon} - \Delta\right]^{-1} \\ &\approx \left[1 + \left(A^\dagger - P_{m,\epsilon}\right)^{-1} \Delta\right] \left(A^\dagger - P_{m,\epsilon}\right)^{-1} \ .\end{aligned} \tag{8}$$

This leads us to the following procedure of solving $D_n X = \eta$:

(1) first solve

$$\left(A^\dagger - P_{m,\epsilon}\right) Y = \eta \; ; \tag{9}$$

(2) then solve

$$\left(A^\dagger - P_{m,\epsilon}\right) X_0 = \eta + \Delta Y \; ; \tag{10}$$

(3) use X_0 as a starting vector to finally solve

$$\left(A^\dagger - P_{n,\epsilon}\right) X = \eta \; . \tag{11}$$

The generation of the starting vector X_0 in steps (1) and (2) is only a small overhead. In fig. 3 we plot the relative residuum $\epsilon^2_{\text{stop}} = \|D_n X - \eta\|^2 / \|X\|^2$ as a function of the number of applications of D_n. In this case $n = 100$ and $m = 30$. We show the number of applications of the matrix D_n for the case of a random starting vector (dotted line) and the case where X_0 was generated according to the above procedure (solid line). The gain is of approximately a factor of two.

Scheme (b)

In the second approach, we use a sequence of polynomials to solve $D_n X = \eta$. To this end we first solve

$$\left[A^\dagger - P_{m_1,\epsilon}\right] X_1 = A^\dagger \eta \tag{12}$$

by choosing a polynomial $P_{m_1,\epsilon}$ and a stopping criterion for the solver $\epsilon^{(1)}_{\text{stop}}$ such that

$$m_1 < n, \; \epsilon^{(1)}_{\text{stop}} > \epsilon_{\text{stop}} \; . \tag{13}$$

The value of $\epsilon^{(1)}_{\text{stop}}$ is chosen such that it is roughly of the same order of magnitude as the error that the polynomial of degree m_1 itself induces. The solution X_1

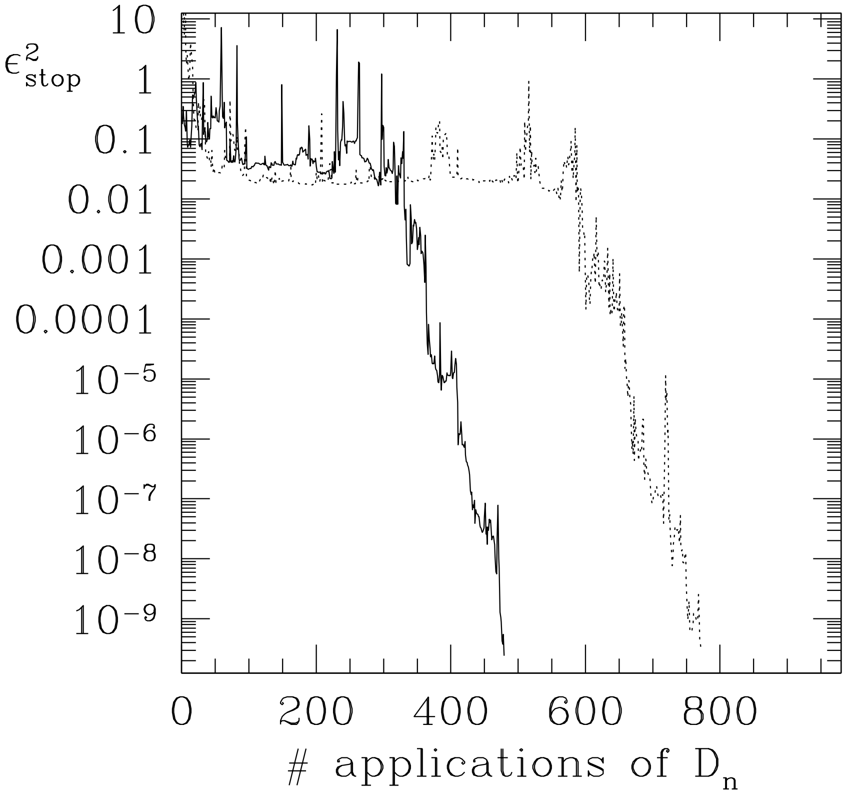

Figure 3: The residuum as a function of the number of applications of the matrix D_n. The dotted line corresponds to a random starting vector. The solid line to a starting vector generated following scheme (a).

is then used as a starting vector for the next equation, employing a polynomial $P_{m_2,\epsilon}$ and stopping criterion $\epsilon_{\text{stop}}^{(2)}$ with

$$m_1 < m_2 < n,\ \epsilon_{\text{stop}}^{(1)} > \epsilon_{\text{stop}}^{(2)} > \epsilon_{\text{stop}}\ . \tag{14}$$

This procedure is then repeated until we reach the desired polynomial $P_{n,\epsilon}$ and stopping criterion ϵ_{stop} to solve the real equation

$$\left[A^\dagger - P_{n,\epsilon}\right] X = A^\dagger \eta\ . \tag{15}$$

As for scheme (a), we gain a factor of about two in the numerical effort. We finally remark that some first tests using the scheme proposed in [13] resulted in a similar performance gain as the two schemes presented above.

In table 1 we give a typical example of the expense of a simulation following strategy (*ii*). We list both the cost of computing the lowest eigenvalue of $D_n^\dagger D_n$ in terms of the number of iterations to minimize the Ritz functional [19] and the number of iterations to solve $D_n X = \eta$. In both applications, a polynomial of degree n is used to approximate the square root. The numbers in table 1 indicate that a quenched calculation, employing Neuberger's operator, leads to a computational cost that is comparable with a dynamical simulation using conventional operators.

N	n	n_{ev}	n_{invert}
8	190	170	80
10	250	325	200
12	325	700	300

Table 1: N is the number of lattice sites along a side of the hypercube; n, the degree of polynomial; n_{ev}, the number of iterations required to obtain the lowest eigenvalue of $D_n^\dagger D_n$; and n_{invert}, the number of iterations necessary to compute $X = D_n^{-1}\eta$.

5 Conclusions

The theoretical advance that an exact chiral symmetry brings to lattice gauge theory is accompanied by the substantial increase in numerical effort that is required to implement operators satisfying the GW relation. Thus, while the Nielsen–Ninomiya theorem has been circumvented, the "no free lunch theorem" has not. Whether alternative formulations, such as domain wall fermions, can help in this respect remains to be seen.

Appendix

We give here the explicit definitions needed in eq. (1). The forward and backward derivatives ∇_μ, ∇_μ^* act on a vector $\Phi(x)$ as

$$\begin{aligned} \nabla_\mu \Phi(x) &= \frac{1}{a}\left[U(x,\mu)\Phi(x+a\hat{\mu}) - \Phi(x)\right] \\ \nabla_\mu^* \Phi(x) &= \frac{1}{a}\left[\Phi(x) - U(x-a\hat{\mu},\mu)^{-1}\Phi(x-a\hat{\mu})\right] \ , \end{aligned}$$

where $\hat{\mu}$ denotes the unit vector in direction μ. The (gauge) field $U(x,\mu) \in SU(3)$ lives on the links connecting lattice points x and $x + a\hat{\mu}$ and acts on the colour index $\alpha = 1,2,3$ of the field Φ. Finally, the Dirac matrices γ_μ, $\mu = 0,1,2,3$ are hermitean 4×4 matrices acting on the Dirac index i of the field Φ. Their explicit form is given by

$$\gamma_\mu = \begin{pmatrix} 0 & e_\mu \\ e_\mu^\dagger & 0 \end{pmatrix} \tag{16}$$

with

$$e_0 = 1, \ \ e_k = -i\sigma_k \tag{17}$$

and

$$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} . \tag{18}$$

With the choice of the γ matrices given above, the matrix $\gamma_5 = \gamma_0\gamma_1\gamma_2\gamma_3$ is diagonal and given by

$$\gamma_5 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} . \tag{19}$$

We finally note that whenever repeated indices appear, they are summed over.

References

[1] K.G. Wilson, Phys. Rev. **D10** (1974) 2445.

[2] P. Hasenfratz, V. Laliena and F. Niedermayer, Phys.Lett.**B427** (1998) 125.

[3] P. Hasenfratz, Nucl. Phys. **B525** (1998) 401.

[4] H. Neuberger, Phys. Lett. **B417** (1998) 141 and **B427** (1998) 353.

[5] M. Lüscher, Phys. Lett. **B428** (1998) 342.

[6] S. Chandrasekharan, Phys. Rev. **D60** (1999) 074503.

[7] P. Hasenfratz, Nucl. Phys. B (Proc.Suppl.) **63A-C** (1998) 53.

[8] P.H. Ginsparg and K.G. Wilson, Phys. Rev. **D25** (1982) 2649.

[9] F. Niedermayer, Nucl. Phys. B (Proc.Suppl.) **73** (1999) 105.

[10] T. Blum, Nucl.Phys.B (Proc.Suppl.) **73** (1999) 167.

[11] P. Hernández, K. Jansen and L. Lellouch, CERN preprints CERN-TH/99-197, hep-lat/9907022 and CERN-TH/99-273, hep-lat/9909026.

[12] H. Neuberger, hep-lat/9910040.

[13] A. Borici, hep-lat/9910045.

[14] P. Hernández, K. Jansen and M. Lüscher, Nucl. Phys. **B552** (1999) 363.

[15] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical Recipes*, Second Edition, Cambridge University Press, Cambridge, 1992.

[16] Y. Saad, *Iterative methods for sparse linear systems*, PWS Publishing Company, Boston, 1996.

[17] L. Fox and I. B. Parker, *Chebyshev polynomials in numerical analysis*, Oxford University Press, London, 1968.

[18] R.G. Edwards, U.M. Heller and R. Narayanan, Phys. Rev. **D59** (1999) 094510.

[19] B. Bunk, K. Jansen, M. Lüscher and H. Simma, DESY report (September 1994); T. Kalkreuter and H. Simma, Comp.Phys.Comm. **93** (1996) 33.