INVERSION OF THE FERMIONIC MATRIX AND MULTIGRID

T. BARCZYK AND R. WIT

Institute of Physics, Jagellonian University
Reymonta 4, 30-059 Kraków, Poland

(Received June 5, 1991; revised version received June 24, 1991)

Numerical simulations indicate that the multigrid method has some natural limitations when applied to the calculation of the quark propagator in the Lattice QCD.

PACS numbers: 12.38.Gc

1. Introduction

Since we do not know the analytical solution of the continuos gauge theories [1] we have to use the computer simulations for the lattice QCD calculations. One of the main numerical problems there is the computation of the inverse of the fermionic matrix. In terms of the Dirac matrices γ_μ and the gauge fields U_{ij} this matrix has the following block structure:

$$\begin{aligned} A_{ii} &= I, \\ A_{ij} &= -K(1-\gamma_\mu)U_{ij}, \quad j = i+\vec{\mu}, \\ A_{ij} &= -K(1+\gamma_\mu)U_{ij}, \quad j = i-\vec{\mu}, \\ A_{ij} &= 0, \qquad\qquad\quad \text{otherwise.} \end{aligned}$$

Here K denotes the hopping parameter

$$K = \frac{1}{8+2ma},$$

where m is the quark mass and a is the elementary lattice length.

In what follows the cases with

$$K < K_{cr} = \tfrac{1}{8}$$

are considered. The relation $j = i + \vec{\mu}$ means that the site j is next to the site i in the μ-direction. To calculate the inverse of the matrix A means that we have to find N times the solution of the following algebraic problem

$$Ax = b,$$

$$x \in C^N, \qquad b \in C^N, \qquad A \in C^{N*N}.$$

Different techniques have been introduced to solve this problem in an efficient way. The most promising ones seem to be the incomplete LU decomposition [2], introduced in a different context by Meijerink and van der Vorst (*cf.* the papers contained in [3]), and the M(ulti) - G(rid) approach (*cf.* [4]) with a variety of iterative basic algorithms. Our research was triggered by the following observation.

The C(onjugate) G(radient) method is known to be one of the most efficient local algorithms. When used to solve an standard elliptic problem, it shows, however, a rather peculiar behaviour [4]. The residuum norm reduction for initial iterations is not particularly fast, for subsequent iterations the error norm could even increase and only then, after the appropriate minimization subspace is settled, one observes a sudden residuum norm fall-off by several orders of magnitude. In Oyanagi's [2] numerical data presentation, concerning the inversion of the fermionic matrix, the CG method was also the worst one for the first, say, 200 iterations.

On the other hand, a good performance of the MG with the, *e.g.* G(auss)–S(eidel) method, is well understood. If we take the Fourier transform of the errors, GS supresses the high frequencies, the coarse grid takes care of the lower ones. Therefore it is rather natural to ask if the CG method, put on the MG, would improve the convergence at the early iterations, preserving its sometimes striking performance at the end of the calculations. Theoretical analysis does not seem to give us clear suggestions in this case (*cf.* [3]).

Note that the preconditioning LU method leads to a steep residuum norm decrease right from the beginning of the iteration procedure.

2. Implementation

Due to the resource memory shortage (we had at our disposal only 2.5 Mb on the CROMEMCO CS – 420 computer) we were forced to limit ourselves in the following way. We resigned from the gauge structure right at the beginning and therefore we do not touch here the problem of the gauge invariance . For the MG calculations we used a two level $6^4 - 3^4$ scheme with the spin structure retained. As a consequence we had to deal with complex vectors. The incomplete LU decomposition requires at least 4

grid points in one space — time dimension. This requirement is connected with the periodic boundary conditions implemented in our algorithms. The 4^4 coarse grid for the incomplete decomposition implied an 8^4 (dense) grid for this acceleration method. Therefore at the beginning of our calculations we tested this method only for scalar fields.

Our program was written in the *C* language as a rather natural consequence of using the UNIX operating system. The results presented below, if not stated otherwise, are coming from 4 iterations on the dense grid and 10 relaxation sweeps on the coarse one.

The most time consuming operation is the matrix times vector multiplication. For comparison between different methods we use this multiplication as a measure of the number of iterations.

As well known a simple *MG* algorithm has the following structure:

1. Perform a certain number of iterations (*e.g.* *GS*) on the dense grid to get a rough approximation for the solution of the initial equation $A\vec{x} = \vec{b}$.
2. Project the fine grid values of the residual vector $\vec{r}$ on the coarse lattice.
3. Solve approximately the auxilliary equation $A\vec{e} = \vec{r}$ on a coarse sublattice.
4. Extrapolate these coarse sublattice values of the error vector $\vec{e}$ to the fine grid and correct $\vec{x}$.
5. Return to the first step.

As a starting value for $\vec{x}$ (on the dense grid) we used $(LU)^{-1}_{\text{inc}}\vec{b}$, on the coarse grid we commenced with the error vector $\vec{e} = \vec{0}$.

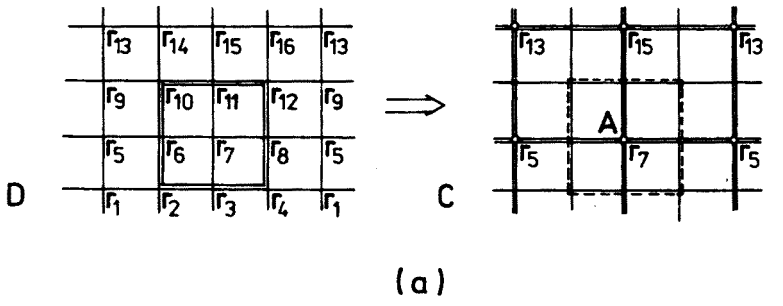
One of the most important ingredients of the *MG* acceleration is the transport of the solution values between the coarse and dense grids. In four dimensions an elementary cube containes 81 knots. Tab. I shows their characteristics relevant for our calculations.

TABLE I

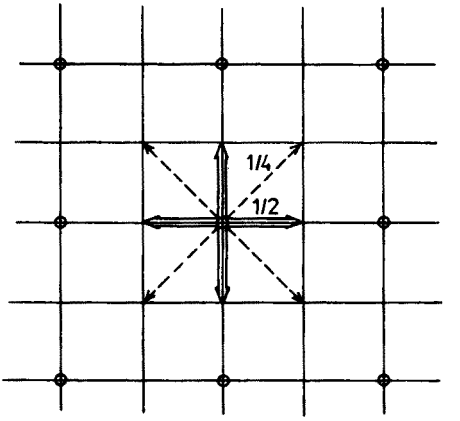
Distances and relative weights for a 4-dimensional hypercube

Distance from the centre	Weight	Number of knots
0	1	1
a	1/2	8
$a\sqrt{2}$	1/4	24
$a\sqrt{3}$	1/8	32
$2a$	1/16	16

For example, when moving back from the coarse to the dense grid every



(a)



(b)

Fig. 1. (a) transmission of residuum values from the dense to the coarse grid, (b) scattering from coarse to the dense grid values with appropriate weights.

point retains the appriopriate solution value, propagates $\frac{1}{2}$ of it to the 8 nearest neighbours, $\frac{1}{4}$ to the next 24 neighbours etc. An inclusion of an overall normalisation factor is obvious. Similar procedure was adopted for the return journey to the coarse grid.

It is easy to demonstate this procedure on a two - dimensional lattice. In Fig. 1a the transport of the dense grid residuum values to the coarse one, according to the formula

$$r_A = \left(r_7 + \tfrac{1}{2}(r_3 + r_6 + r_{11} + r_8) + \tfrac{1}{4}(r_2 + r_{10} + r_{12} + r_4)\right)/4$$

is shown.

Since in our calculations the *CG Least Norm* and *Least Square* results did not differ very much we present here only the numbers concerning the *LN* approach.

3. Results

The typical behaviour of the *CG* method for different values of the hopping parameter K is shown in Fig. 2.

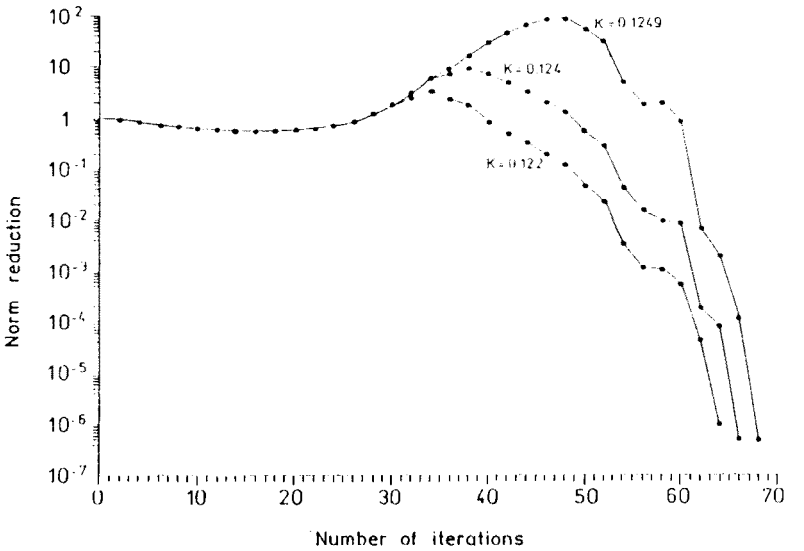


Fig. 2. Norm reduction of the *CG Least Norm* method for different values of the hopping parameter K for a 6^4 grid

The final results are almost independent of the K values but one has to be cool-headed before the real convergence is seen. Putting the *CG* method on *MG* does not help: the convergence rate is better for the initial iterations, but eventually the original *CG* finds its way to the drastic residuum norm reduction (Fig. 3). It turns out that *MG* disturbs the construction of the set of independent vectors which is crucial for the convergence of the *CG* method.

This is in agreement with different, 2-dimensional calculations when non-trivial gauge structure is included [5]. However the preconditioning with fine tuning (good acceleration parameter choice) could lead to a much better residuum norm reduction than *MG*, what can be seen in Fig. 4.

Only recently we were able to check our calculations with the complex variables on the 10^4 vs 5^4 grids. None of the conclusions presented

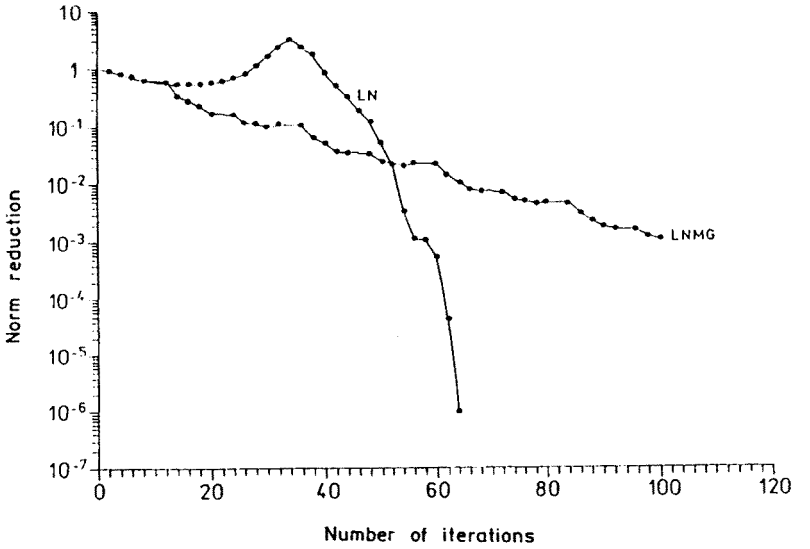


Fig. 3. Norm reduction for the *CG LN* with and without *MG* acceleration

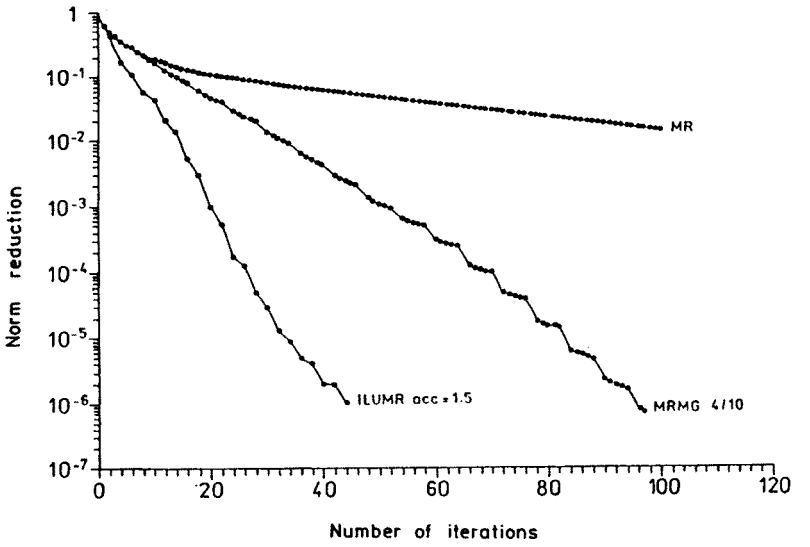


Fig. 4. A comparison of three different iteration procedures with M(inimal) R(residual) and *MR* accelerated by the *MG*, and *ILU* decomposition (best acceleration parameter choice)

above has to be changed. This is indicating again that the *MG* approach

is not very helpfull in the QCD lattice calculations, when combined with the CG iteration prescription, and that the ILU technique may lead to faster convergence of the iterative calculations than those based on the MG philosophy.

Useful discussions with J. Wosiek at the early stage of this work are acknowledged.

REFERENCES

[1] J. Wosiek, *Acta Phys. Pol.* **B18**, 519 (1987).
[2] Y. Oyanagi, *Com.Phys.Comp.* **42**, 333 (1986).
[3] H.A. van der Vorst, *Preconditioning by Incomplete Decompositions*, Academisch Computer Centrum, Utrecht 1982
[4] J.R. Kightley, C.P. Thompson, *On the Performance od Some Rapid Elliptic Solvers On a Vector Processor*, Computer Science and Systems Division, Harwell Laboratory, Harwell 1986.
[5] R.C. Brower, C. Rebbi and E. Vicari, Projective Multigrid for Propagators in Lattice Gauge Theory, Boston University preprint, BU-HEP /90/16, Boston 1990.