Journal of Statistical Physics, Vol. 107, Nos. 1/2, April 2002 (© 2002)

A Multigrid-Solver for the Discrete Boltzmann Equation

Jonas Tölke,[1] Manfred Krafczyk,[1] and Ernst Rank[2]

Received February 14, 2001; accepted November 30, 2001

This paper introduces a nonlinear multigrid solution approach for the discrete Boltzmann equation discretized by an implicit second-order Finite Difference scheme. For simplicity we restrict the discussion to the stationary case. A numerical example shows the drastically improved efficiency in comparison to the widely used Lattice–Bathnagar–Gross–Krook (LBGK) approach.

KEY WORDS: Lattice Boltzmann method; multigrid.

1. INTRODUCTION

For more than a decade an explicit numerical scheme for the discrete Boltzmann equation, the so-called Lattice–Bathnagar–Gross–Krook (LBGK) approach[1] has been used quite successfully to obtain approximate solutions for weakly compressible and incompressible Navier–Stokes problems. Although LBGK models have shown to be effective simulation tools for a variety of such problems especially for the transient regime, the benefits of state-of-the-art numerical techniques for the solution of the discrete Boltzmann equation have not been sufficiently explored. A first step in this direction is to decouple the spatial and temporal discretization by using Finite Difference[2, 3]—or Finite Volume-based methods[4] on non-uniform grids. Beside the possibility to utilize a Galerkin-based FD-solution approach[5] one can furthermore obtain solutions by implicit schemes as described in.[6] These have been successfully applied to high Reynolds-number flows with moderate numbers of degrees of freedom (DOF). As an example for an

[1] Institut für Computeranwendungen im Bauwesen, Technische Universität Braunschweig, Germany; e-mail: toelke@cab.bau.tu-bs.de

[2] Lehrstuhl für Bauinformatik, Technische Universität München, Germany.

0022-4715/02/0400-0573/0 © 2002 Plenum Publishing Corporation

advanced numerical approach suitable for a huge number of DOF this paper introduces a multigrid solution approach for the stationary discrete Boltzmann equation.

2. ITERATIVE SOLUTION METHODS

Let

$$\mathscr{L}(\mathbf{u}) = F \tag{1}$$

be a nonlinear equation system resulting from the discretization of a (partial) differential equation. Basically all iterative numerical solution schemes for Eq. (1) can be cast to the following form:

$$\mathsf{P}\frac{\Delta\mathbf{u}^m}{\omega} = -\mathbf{d}^m \tag{2}$$

Here $\mathbf{d}^m = \mathscr{L}(\mathbf{u}^m) - F$ is the so-called defect. The iteration error is the difference between the exact solution $\mathbf{u}^\infty$ of the discrete equations and the actual approximation $\mathbf{u}^m$ and will be simply referred to as the *error* in the course of the discussion to follow and has to be distinguished from the discretization error due to finite grid resolution. P is a suitable preconditioner and ω a relaxation parameter controlling the convergence behavior of the scheme.

An improved solution is obtained from

$$\mathbf{u}^{m+1} = \mathbf{u}^m + \Delta\mathbf{u}^m \tag{3}$$

If $\mathsf{J}(m)$ is the Jacobian of $\mathscr{L}$ with respect to $\mathbf{u}^m$:

$$\mathsf{J}(m) = \frac{\partial\mathscr{L}^{(i)}}{\partial u_j}(\mathbf{u}^m) \tag{4}$$

it can be shown,[7] that the iterative scheme converges to the solution $\mathbf{u}^\infty$ if the spectral radius $\rho(\mathsf{M})$ of the amplification matrix $\mathsf{M}(m)$ given by

$$\mathsf{M}(m) := \mathsf{1} - \omega\mathsf{P}^{-1}\mathsf{J}(m) \tag{5}$$

is smaller than unity for $m \to \infty$. The spectral radius $\rho(\mathsf{M})$ of a matrix is the largest eigenvalue λ of M:

$$\rho(\mathsf{M}) = \max\{|\lambda_j|,\ i = 1,\ldots, n\} \tag{6}$$

The value $\lim_{m\to\infty} \rho(\mathsf{M}(m))$ is called the convergence rate of the iterative scheme because it characterizes the *asymptotic* behavior of the iteration error. In addition, we introduce a suitable norm characterizing the convergence behavior of the error because fast solvers converge within a few iterations and thus never show asymptotic behavior characterized by the spectral radius. The spectral norm of a matrix M is defined as

$$\|\mathsf{M}\| = \sqrt{\rho(\mathsf{M}^T\mathsf{M})} \tag{7}$$

and is often referred to as the contraction number because it determines the error reduction of the iterative process:

$$\|\mathbf{u}^{m+1} - \mathbf{u}^{\infty}\| \leqslant \|\mathsf{M}(m)\| \, \|\mathbf{u}^{m} - \mathbf{u}^{\infty}\| \tag{8}$$

3. THE MULTIGRID APPROACH

3.1. Motivation

For classical iteration techniques like Point–Jacobi or Point–Gauss–Seidel the rate of convergence depends on the mesh resolution h:

$$\rho \sim 1 - O(h^2) \tag{9}$$

Thus a fine mesh resolution results in a very slow convergence and the number of iterations required to solve the system implies a huge computational effort for large equation systems. As the LBGK scheme can be identified as a Point–Jacobi-type scheme it shows the corresponding behavior as will be demonstrated below.

The underlying reason for the weak convergence rate of standard iterative schemes for large numbers of DOF becomes obvious when looking at the evolution of the wavenumber spectrum of the error. For fine grids damping of small wavenumbers is (in contrast to the large ones) very weak so that a huge number of iterations is required until the (long wavelength part of the) error amplitude is damped and convergence is reached. The basic idea of a multigrid approach is to represent the low-frequency part of the error on a coarser grid with a reduced number of DOF. After smoothing or eliminating the error on the coarse grid, an improved solution is transferred back to the fine grid where remaining high frequency parts of the error are efficiently damped by additional fine grid iterations. The transfer process from the fine grid to the coarse grid is the so-called restriction, the opposite transfer is referred to as prolongation. Using a multigrid scheme the number of iterations can be shown to become

independent of the number of unknowns provided that certain mathematical properties of the restriction and prolongation as well as the iterative scheme itself are fulfilled.

3.2. The Basic Linear Multigrid Procedure

We now sketch a linear twogrid algorithm LTG for a system of *linear* equations, resulting from the discretization of a partial differential equation on the grid level l. The system is written in matrix form as

$$L_l u_l = F_l \tag{10}$$

Let u_l' be an approximation for the exact solution u_l of the discrete equations. On the fine grid we perform ν_1 smoothing steps using a smoothing operator $(S)_l$:

$$\bar{u}_l = (S)_l^{\nu_1}\ (u_l', F_l) \tag{11}$$

Now the error

$$v_l = \bar{u}_l - u_l \tag{12}$$

is smooth. The exact correction v_l can be computed as follows.

$$L_l(\bar{u}_l - v_l) = F_l \tag{13}$$

Since the problem is linear, we have:

$$L_l v_l = L_l \bar{u}_l - F_l = d_l \tag{14}$$

The solution of (14) is as difficult as the original problem (10). Since the error is smooth, we can solve the problem on a coarser grid. The defect $d_l = L_l \bar{u}_l - F_l$ is transferred to the coarse grid via the restriction operator r, which will be defined in Section 4.4:

$$d_{l-1} = r d_l \tag{15}$$

The exact correction v_{l-1} on the coarse grid is computed with

$$L_{l-1} v_{l-1} = d_{l-1} \tag{16}$$

where L_{l-1} is the matrix representing the difference equations on grid level $l-1$. The exact correction is then mapped back onto the fine grid using a prolongation operator p, which will be defined in Section 4.4:

$$\hat{u}_l = \bar{u}_l - pv_{l-1} \tag{17}$$

After the correction the error is not smooth anymore an we can perform ν_2 smoothing steps:

$$\tilde{u}_l = (S)_l^{\nu_2}\ (\hat{u}_l, F_l) \tag{18}$$

In order for the convergence rate to be independent of the number of DOF, the restriction and prolongation operator must obey certain constraints[8] (Section 4) and the operator $(S)_l$ must have the smoothing property.[9] A single iteration step of the linear twogrid algorithm in pseudo-code is thus given by

Algorithm 1. Linear twogrid algorithm LTG

```
Procedure LTG (integer l, array u, array F)
array d,v

u = (S)_l^{ν1} (u, F)   //pre-smoothing
d = r(L_l u − F)        //restriction of defect
v = L_{l-1}^{-1} d      //exact correction
u = u − pv              //prolongation of correction
u = (S)_l^{ν2} (u, F)   //post-smoothing
```

These twogrid-iterations are performed until a suitable norm of the defect indicates convergence:

Algorithm 2. Outer twogrid loop

```
do
  call LTG(l, u_l, F_l)      //Twogrid-iteration
  d = L_l u − F              //defect
  if (‖d‖ < eps) stop        //stopping criterion
enddo
```

3.3. Extension to Multiple Grids

For a large number of DOF the computation of the exact correction on the coarse grid is still cumbersome because the matrix to invert is still

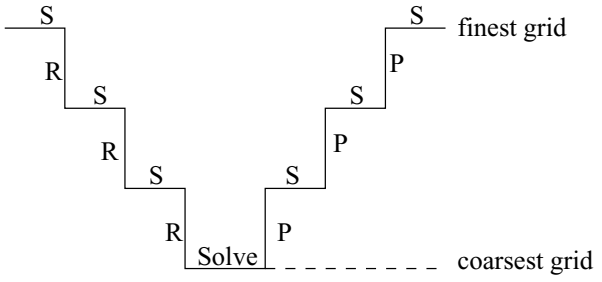

Fig. 1. V-cycle.

large. The idea is to compute an *approximate* solution on the coarse grid. This approximate solution together with a new coarser grid is then used for another twogrid-iteration. This approach is recursively repeated until the coarsest grid allows an exact solution, leading to the so-called V-cycle of the multigrid method (see Fig. 1).

This linear multigrid (LMG) algorithm exhibits a natural recursivity:

Algorithm 3. Linear multigrid algorithm LMG

```
procedure LMG (integer l, array u, array F)
array d, v
  if (l == 0) then
      u = L_0^{-1} F                    //exact correction
  else
      u = (S)_l^{ν1} (u, F)             //pre-smoothing
      d = r(L_l u − F)                  //restriction of defect
      v = 0                             //initialization of correction
      call LMG(l − 1, v, d)             //recursive call of LMG
      u = u − pv                        //prolongation of correction
      u = (S)_l^{ν2} (u, F)             //post-smoothing
  endif
```

Again these multigrid-iterations are performed applying Algorithm 2 where LTG is replaced by LMG until convergence.

4. A MULTIGRID METHOD FOR THE DISCRETE BOLTZMANN EQUATIONS

4.1. Discretization of the Stationary Discrete Boltzmann Equation

The differential equation to solve is given by the stationary discrete Boltzmann equation using a BGK approximation for the collision operator and the discretized velocity space of the so-called incompressible *d2q9*-model:[1, 10]

$$\xi_a \cdot \frac{\partial f_a}{\partial \mathbf{x}} + \frac{1}{\tau}(f_a - f_a^{(0)}) = 0 \qquad a = 0,\ldots, 8 \tag{19}$$

with velocity vectors

$$(\xi_a)_{a=0,\ldots,8} = c \begin{pmatrix} 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \end{pmatrix} \tag{20}$$

and equilibrium distributions

$$f_a^{(0)} = t_a \left\{ p + p_0 \left(\frac{\xi_{a\alpha} u_\alpha}{c_s^2} + \frac{u_\alpha u_\beta}{2c_s^2} \left(\frac{\xi_{a\alpha}\xi_{a\beta}}{c_s^2} - \delta_{\alpha\beta} \right) \right) \right\} \tag{21}$$

Here $t_0 = \frac{4}{9}$, $t_1 = t_2 = t_3 = t_4 = \frac{1}{9}$, $t_5 = t_6 = t_7 = t_8 = \frac{1}{36}$, τ is the relaxation time and c is a parameter, which determines the speed of sound $c_s = \sqrt{\frac{c^2}{3}}$. The value p_0 is a reference pressure. Note that the summation convention is used with respect to Greek letters only, defining, e.g., $\xi_{a\alpha} u_\alpha = \sum_{\alpha=1}^{2} \xi_{a\alpha} u_\alpha$. It can be shown[10] that for small Mach numbers the moments given by

$$p = \sum_a f_a \qquad p_0 \mathbf{u} = \sum_a \xi_a f_a \tag{22}$$

are solutions of the incompressible stationary Navier–Stokes equation

$$\nabla \cdot \mathbf{u} = 0 \tag{23}$$

$$\mathbf{u} \cdot \nabla \mathbf{u} = -\frac{c_s^2}{p_0} \nabla p + \nu \nabla^2 \mathbf{u} \tag{24}$$

if the kinematic viscosity is related to the microscopic relaxation time by $\nu = \frac{\tau}{3} c^2$. The LBGK approach is to discretize Eq. (19) by

$$f_a(\mathbf{x} + \Delta t \xi_a, t + \Delta t) - f_a(\mathbf{x}, t) = -\frac{\Delta t}{\tau}(f_a(\mathbf{x}, t) - f_a^{(0)}(\mathbf{x}, t)) \qquad a = 0,\ldots, 8 \tag{25}$$

where $\mathbf{e}_a = \xi_a / c$ are the discrete lattice vectors. For this approach the order of the scheme can be improved by identifying the viscosity as $\nu = \frac{2\tau - \Delta t}{6} c^2$.

The discretization for the multigrid approach uses a second order upwind FD scheme in the flow domain and a first order scheme at the first layer from the the boundaries. The Finite Differences are build along the characteristics of the system.

First order upwind differences yield

$$\frac{c}{\Delta x}\left(f_a(x_i, y_j) - f_a(x_i - \Delta x e_{a1}, y_j - \Delta x e_{a2})\right) = \xi_a \cdot \frac{\partial f_a}{\partial \mathbf{x}} + O(\Delta x) \tag{26}$$

and second order upwind differences give

$$\frac{c}{\Delta x}\left(\frac{3}{2} f_a(x_i, y_j) - 2 f_a(x_i - \Delta x e_{a1}, y_j - \Delta x e_{a2}) + \frac{1}{2} f_a(x_i - 2\Delta x e_{a1}, y_j - 2\Delta x e_{a2})\right) = \xi_a \cdot \frac{\partial f_a}{\partial \mathbf{x}} + O(\Delta x^2) \tag{27}$$

The resulting difference equations are

$$f_a(i, j) - f_a(i - e_{a1}, j - e_{a2}) + \frac{\Delta x}{c\tau}\left(f_a(i, j) - f_a^{(0)}(i, j)\right) = 0 \tag{28}$$

and

$$3 f_a(i, j) - 4 f_a(i - e_{a1}, j - e_{a2}) + f_a(i - 2e_{a1}, j - 2e_{a2}) + \frac{2\Delta x}{c\tau}\left(f_a(i, j) - f_a^{(0)}(i, j)\right) = 0 \tag{29}$$

where $f(i, j) := f(x_i, y_j)$ and $f(i - e_{a1}, j - e_{a2}) := f(x_i - \Delta x e_{a1}, y_j - \Delta x e_{a2})$. Dirichlet boundary conditions for the velocity field can be set as follows: For every distribution f_a pointing out of the computational grid and for f_0, Eq. (28) can be applied, whereas for the other distributions the modified bounce-back scheme of Ladd[11, 12] is used. This defines the values for 8 of 9 distributions. One additional equation can be obtained by linear extrapolation of the pressure. An example is given in Section 5.

4.2. The Nonlinear Multigrid Extension

As the discretized Boltzmann equations are nonlinear, we choose a corresponding nonlinear multigrid extension based on,[13] to solve the nonlinear equation system

$$\mathcal{L}_l(u_l) = F_l \tag{30}$$

resulting from Eqs. (28), (29), and the corresponding boundary conditions.

Let $\mathscr{L}_l$ be the nonlinear operator with respect to a given grid level and u_l' be an approximate solution for Eq. (30). As in the linear case we perform ν_1 smoothing steps using the smoothing operator $(S)_l$ to obtain $\bar{u}_l$ and a smooth error v_l. The exact correction v_l is then

$$\mathscr{L}_l(\bar{u}_l - v_l) = F_l \tag{31}$$

which is to be approximately computed by the coarse grid correction. Since the problem is nonlinear, we write

$$\mathscr{L}_l(\bar{u}_l - v_l) - \mathscr{L}_l(\bar{u}_l) = F_l - \mathscr{L}_l(\bar{u}_l) = -d_l \tag{32}$$

Now the problem is transferred to the coarse grid using the restriction operator r:

$$\mathscr{L}_{l-1}(u_{l-1}) - \mathscr{L}_{l-1}(r\bar{u}_l) = -rd_l \tag{33}$$

Equation (33) can be written as:

$$\mathscr{L}_{l-1}(u_{l-1}) = \mathscr{L}_{l-1}(r\bar{u}_l) - rd_l = F_{l-1} \tag{34}$$

In contrast to the linear scheme one solves (34) for u_{l-1} and not just for the correction v_{l-1}. After Solving (34), the coarse grid correction v_{l-1} can be computed with

$$v_{l-1} = r\bar{u}_l - u_{l-1} \tag{35}$$

The correction is now transferred back to the fine grid using the prolongation operator p

$$\hat{v}_l = pv_{l-1} \tag{36}$$

Now we can correct the solution $\bar{u}_l$ with

$$\hat{u}_l = \bar{u}_l - \hat{v}_l \tag{37}$$

and then perform ν_2 post-smoothing steps using (18).

Using Taylor expansion around $\bar{u}_l$ in Eq. (32), the desired exact correction can be written as

$$v_l = L_l^{-1}(\bar{u}_l)\, d_l + O(\|v_l\|^2) \tag{38}$$

where $L_l(\bar{u}_l)$ is the Jacobian at the point $\bar{u}_l$. Using Taylor expansion, Eq. (36) can be written as

$$\hat{v}_l := pL_{l-1}^{-1}(r\bar{u}_l)\, rd_l + O(\|rd_l\|^2) \tag{39}$$

Comparison of the desired correction (38) and the computed correction (39) shows that $v_l \approx \hat{v}_l$, if the terms of second order are small (indicating that a good start vector is needed) and v_l is smooth.

The nonlinear multigrid algorithm can be written as:

Algorithm 4. Nonlinear multigrid algorithm NMG

```
procedure NMG (integer l, array u, array F)
array F_c,u_c
  if (l == 0) then
    u = L_0^{-1} F                          //exact solution
  else
    u = (S)_l^{ν1} (u, F)                   //pre-smoothing
    F_c = L_{l-1}(ru) - r(L_l(u) - F)       //right hand side of the coarse grid
    u_c = ru                                //initialization of coarse grid solution
    call NMG(l-1, u_c, F_c)                 //recursive call of NMG
    u = u + p(u_c - ru)                     //prolongation of correction
    u = (S)_l^{ν2} (u, F)                   //post-smoothing
  endif
```

4.3. Computing the Start Vector by Nested Iteration

As discussed in Section 4.2 the nonlinear multigrid approach requires a good initial solution to work efficiently. Such a start vector can be obtained by a so-called nested iteration. The idea is to compute an exact solution on the coarsest grid which is then successively interpolated on the finer grids where this initial solution is improved by one or more multigrid iterations. In Fig. 2 the algorithm is sketched. The interpolated solution on each grid is improved by *one* multigrid iteration.

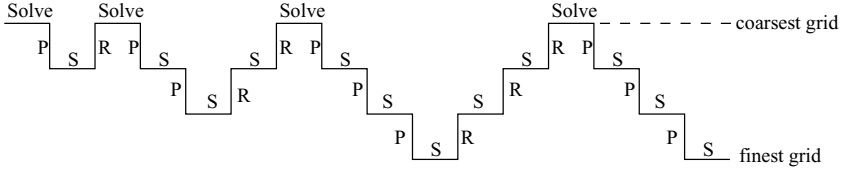

Fig. 2. Nested iteration.

The corresponding pseudo-code reads:

Algorithm 5. Nested iteration

$\tilde{u}_0 = \mathscr{L}_0^{-1}(F_0)$
do $k = 1, l$
  $\tilde{u}_k = \tilde{p}\tilde{u}_{k-1}$
  do $j = 1$, niter
    call NMG(k, $\tilde{u}_k$, F_k)
  enddo
enddo

4.4. Prolongation and Restriction

Prolongation is based on bilinear interpolation including values on the boundaries of the computational domain. For the restriction "full weighting"[9] is used. In stencil notation[17] these operators can be written as:

$$[p] = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix} \qquad [r] = \begin{bmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{bmatrix}$$

4.5. Smoother

We use a *collective* Gauss–Seidel-smoother.[17] At each grid node (i, j) for the nine variables $f_a(i, j)$ a local Newton–Raphson procedure is used to solve the 9×9 equation system while keeping the other variables fixed. In one smoothing operation we use four Gauss–Seidel-iterations with different orderings: Forward, backward, forward vertical line and backward vertical line.[17]

4.6. Measure of MG Efficiency

The efficiency of a multigrid iteration can (in absence of an analytical solution) be evaluated using an appropriate norm of the defect. Each smoothing and correction step should result in a reduction of this norm. The contraction number ζ_j defined as

$$\|\mathbf{d}^{j+1}\| \leqslant \zeta_j \|\mathbf{d}^j\| \tag{40}$$

is a good measure for the error reduction after an iteration. For well-posed problems the spectral norm (7) and the contraction number (40) should behave in a similar manner.

For a given problem several multigrid iterations may be necessary. Accordingly, an averaged contraction number can be defined as

$$\bar{\zeta} = \frac{1}{m} \sum_{j=1}^{m} \zeta_j \tag{41}$$

where m is the number of iterations.

5. NUMERICAL EXAMPLE: CAVITY FLOW

As a numerical example we show results for a lid driven cavity flow for a Reynolds number $\mathrm{Re} = \frac{u_0 L}{\nu} = 10$ obtained by the LBGK approach as well as the multigrid scheme. The problem is depicted in Fig. 3.

The boundary conditions for the upper boundary of this problem are set as discussed in Section 4.1.

$$\frac{1}{c\tau}\left(f_0(i,n) - f_0^{(0)}(i,n)\right) = 0 \tag{42}$$

$$f_2(i,n) - f_2(i,n-1) + \frac{\Delta x}{c\tau}\left(f_2(i,n) - f_2^{(0)}(i,n)\right) = 0 \tag{43}$$

$$f_5(i,n) - f_5(i-1,n-1) + \frac{\Delta x}{c\tau}\left(f_5(i,n) - f_5^{(0)}(i,n)\right) = 0 \tag{44}$$

$$f_6(i,n) - f_6(i+1,n-1) + \frac{\Delta x}{c\tau}\left(f_6(i,n) - f_6^{(0)}(i,n)\right) = 0 \tag{45}$$

$$f_2(i,n) - f_4(i,n) = 0 \tag{46}$$

$$f_1(i,n) - f_3(i,n) - \frac{2}{3}\frac{p_0 u_0}{c} = 0 \tag{47}$$

$$f_7(i,n) - f_5(i,n) + \frac{2}{12}\frac{p_0 u_0}{c} = 0 \tag{48}$$

$$f_8(i,n) - f_6(i,n) - \frac{2}{12}\frac{p_0 u_0}{c} = 0 \tag{49}$$

$$\sum_a f_a(i,n) - 2\sum_a f_a(i,n-1) + \sum_a f_a(i,n-2) = 0 \tag{50}$$

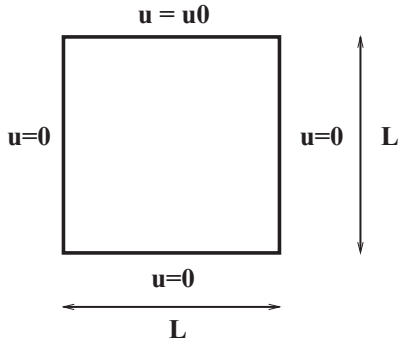

Fig. 3. Setup of the driven cavity problem.

Equations (42)–(45) are the difference Eq. (28). Equations (46)–(49) are the Bounce–Back scheme of Ladd[11, 12] to obtain the desired velocity at the boundary. Equation (50) is the linear extrapolation of the pressure. Note that in contrast to LBGK we can lower the BC velocity error down to machine precision by decreasing the maximum iteration error of the equation solver.

The other boundary conditions are imposed in an equivalent way.

5.1. Error Due to Finite Grid Resolution

The root mean square error $E2$ of the velocity field due to finite grid resolution can be computed with the following formula

$$E2 = \frac{\sqrt{\sum (\mathbf{u}_1 - \mathbf{u}_0)^2}}{\sqrt{\sum \mathbf{u}_2^0}} \tag{51}$$

where $\mathbf{u}_0$ is the reference solution and $\mathbf{u}_1$ is the solution under consideration.

We have computed the stationary flow field for grid resolutions $\Delta x = 2^{-l} \cdot \mathbf{L}$ both with the LBGK ($l = \{5,\ldots, 9\}$) and the Multigrid approach ($l = \{5,\ldots, 11\}$). The root mean square error $E2$ is computed for both methods, whereas in lack of an analytical solution we use the 513×513 grid-solution (LBGK) as the reference solution for the LBGK-Method and the 2049×2049 grid-solution (MG) as the reference solution for the MG-Method. The values for $E2$ and the error reduction for the grid refinement are listed in Table I. For both schemes the convergence rate for the grid refinement can be interpreted as follows: Both schemes are of

Table I. Relative Velocity Error

Grid resolution	33^2	65^2	129^2	257^2	513^2	1025^2
$E2_{MG}$	0.07131	0.03530	0.01342	0.00402	0.00108	0.00028
Error Red.	–	2.02	2.63	3.34	3.71	3.91
$E2_{LBGK}$	0.12664	0.06225	0.02240	0.00677	–	–
Error Red.	–	2.04	2.77	3.31	–	–

second order in space except near the boundary, where we have a first order scheme.

The pressure fields for a resolution of 257×257 nodes are shown in Fig. 4 for the two approaches and are practically indistinguishable.

For this low Reynolds number the interesting part of the solution is not really the position of the main vortex (which coincides for both the LBGK and the multigrid solution to within 2% with reference values given in ref. 15) but the resolution of the secondary vortices at the lower left and right corner of the system. In the corners a self-similar sequence of counter-rotating vortices can be predicted analytically.[14] In Fig. 5a–b the secondary vortices are shown both for the multigrid and the LBGK solution. The position of the vortex center is almost identical. The form of the vortices can be compared to the self-similar solution (see Fig. 5c) to which the LBGK solution shows the closest resemblance.

5.2. Computational Effort

The multigrid computation was done using V-cycles with *no* pre-smoothing and *one* post-smoothing per iteration, since this setup was the most effective. Note that the prolongation and restriction is a very cheap operation in contrast to the smoothing operation, where the solution of small systems of equation is needed. The initial solution was obtained by the nested iteration described above. The iteration was stopped, when the Euclidean norm of the defect was less than 1.0E-10. Table II shows the cell Reynolds number $\mathrm{Re}_c = \frac{Uh}{\nu}$ and the mean value of the contraction number $\bar{\zeta}$ computed in the Euclidean norm. One can see that for the present problem the number of iterations is not only constant (indicating no dependence on the number of unknowns), but even *decreasing* with the grid size, i.e., the number of unknowns, which is due to the decreasing cell Reynolds number.

While the results of the two simulation methods show reasonable agreement, a closer inspection of the effort to obtain the solutions reveals

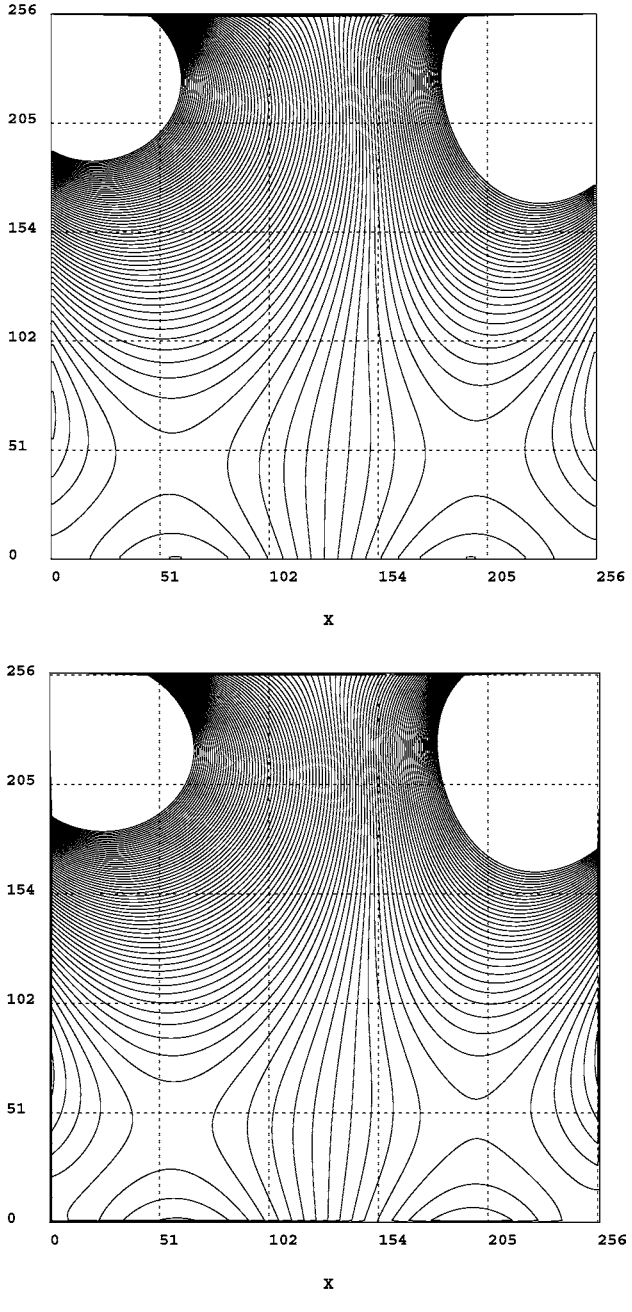

Fig. 4. Isolines for the pressure far field: multigrid- and LBGK-solution.

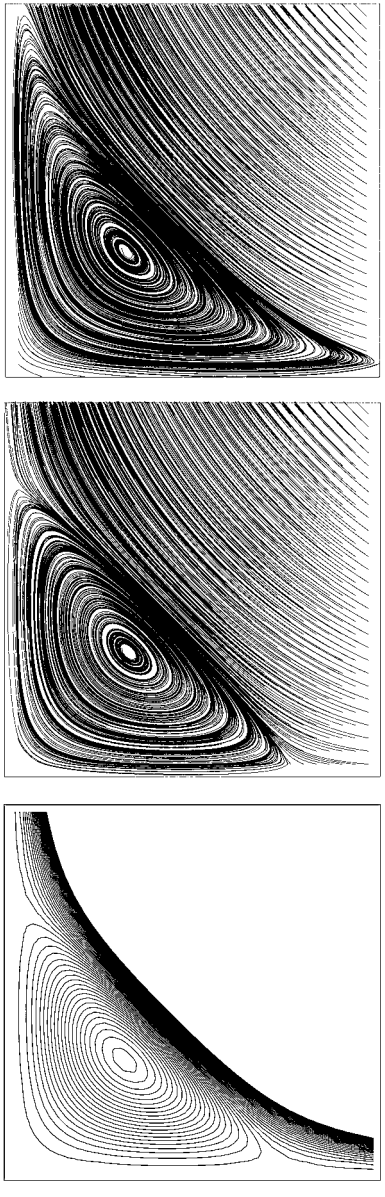

Fig. 5. Lower left vortex: Multigrid-, LBGK-, analytical self-similar solution.

Table II. Contraction Number vs. Grid Size for Re = 10

Grid resolution	33^2	65^2	129^2	257^2	513^2	1025^2	2049^2
Re_c	$\frac{10}{32}$	$\frac{10}{64}$	$\frac{10}{128}$	$\frac{10}{256}$	$\frac{10}{512}$	$\frac{10}{1024}$	$\frac{10}{2048}$
$\bar{\zeta}$	0.50	0.47	0.44	0.41	0.40	0.39	0.37
#Iterations	21	20	19	18	17	17	16

the expected differences. As can be seen in Fig. 6, for finer grids the multigrid approach is substantially faster (for the 1025×1025 and the 2049×2049 grids the LBGK solution times could not even be computed and have been extrapolated from those of the smaller grids).

Although the above results were presented for low Reynolds number flow the nonlinear MG scheme is not restricted to this regime. This can be seen from simulations using a 65 × 65 grid for the driven cavity flow for various Reynolds numbers (see Table III). The start vector was obtained from nested iteration as described above and the iteration was based on V-cycles with no pre- and one post-smoothing step.

From the contraction number it can be seen that the convergence rate decreases for higher Reynolds numbers but still implies a substantial gain in efficiency in comparison to the LBGK scheme.

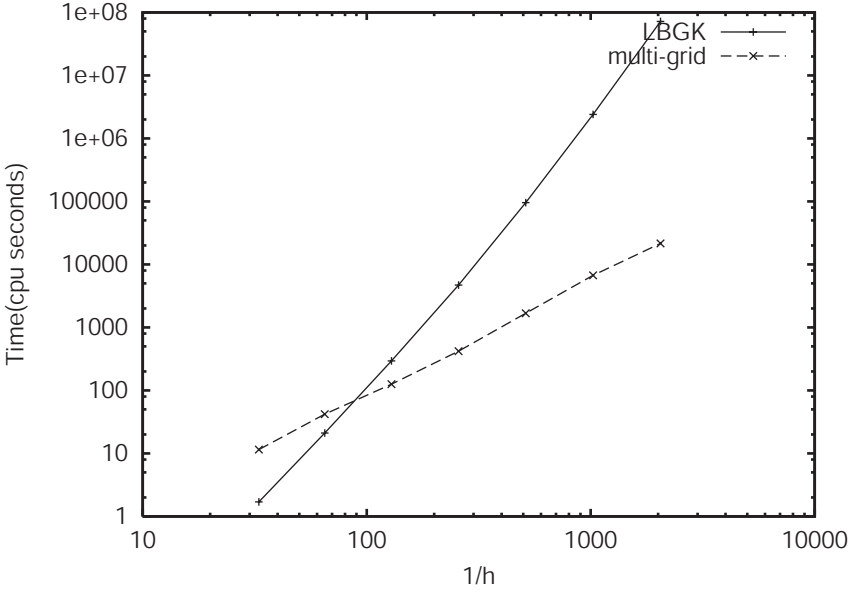

Fig. 6. Comparison of computational efforts for the DC problem using MG and LBGK.

Table III. Contraction Number vs. Reynolds-Number on a 65 × 65 Grid

Reynolds number Re	1.0	10.0	100.0	1000.0
Cell Reynolds number Re_c	$\frac{1}{64}$	$\frac{10}{64}$	$\frac{100}{64}$	$\frac{1000}{64}$
Averaged contraction number $\bar{\zeta}$	0.0031	0.47	0.89	0.98

6. DISCUSSION

The multigrid-solver works efficiently for flows with low and moderate Reynolds numbers. For problems with high Reynolds-numbers further work is required. One approach to improve the efficiency is to use an ordering of grid-points which is related to the local direction of streamlines. This ordering, also called downwind numbering, is constructed using graph-theory.[16]

The procedure outlined here uses prolongation and restriction on a geometrical basis, so the mapping from one level to another is only possible for geometries of moderate complexity. For geometries of very high complexity the so-called algebraic multigrid method[18] is promising.

REFERENCES

1. Y. H. Quian, D. d'Humieres, and P. Lallemand, Lattice BGK models for Navier–Stokes equations, *Europhys. Lett.* **17**(6):479–484 (1992).
2. R. Mei and W. Shyy, On the finite-difference-based lattice Boltzmann method in curvilinear coordinates, *J. Comput. Phys.* **143**:426–228 (1998).
3. D. Kandhai, W. Soll, S. Chen, A. Hoekstra, and P. Sloot, Finite-difference lattice-BGK methods on nested grids, *Comput. Phys. Comm.* **129**:100 (2000).
4. F. Nannelli and S. Succi, The Lattice–Boltzmann equation on irregular lattices, *J. Stat. Phys.* **68**:401–407 (1992).
5. J. Tölke, M. Krafczyk, M. Schulz, and E. Rank, Discretization of the Boltzmann equation in velocity space using a Galerkin approach, *Comput. Phys. Comm.* **129**:91–99 (2000).
6. J. Tölke, M. Krafczyk, M. Schulz, and E. Rank, Implicit discretization and non-uniform mesh refinement approaches for FD discretizations of LBGK Models, *Internat. J. Modern Phys. C* **9**(8):1143–1157 (1998).
7. R. S. Varga, *Iterative Matrix Analysis* (Prentice-Hall, 1962).
8. I. Yavneh, Coarse-grid correction for nonelliptic and singular perturbation problems, *SIAM J. Sci. Comput.* **19**(5):1682–1699 (1998).
9. W. Hackbusch, *Multigrid Methods and Applications* (Springer, 1985).
10. X. He and L. Luo, Lattice-Boltzmann model for the incompressible Navier–Stokes equation, *J. Stat. Phys.* **88**:927–944 (1997).
11. A. J. C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation I, *J. Fluid Mech.* **271**:285 (1994).
12. A. J. C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation II, *J. Fluid Mech.* **271**:311 (1994).

13. A. Brandt, Multi-level adaptive solutions to boundary value problems, *Math. Comp.* **31**:333–390 (1977).
14. H. K. Moffat, Viscous and resitive eddies near a sharp corner, *J. Fluid Mech.* **18**:1–18 (1964).
15. S. Hou, Q. Zou, S. Chen, G. Doolen, and A. Cogley, *J. Comp. Phys.* **118**:329–347 (1995).
16. H. Rentz–Reichert, *Robuste Mehgitterverfahren zur Lösung der Inkompressiblen Navier–Stokes Gleichung*, Ph.D. Thesis (University Stuttgart, 1996).
17. P. Wesseling, *An Introdution To Multigrid Methods* (John Wiley & Sons, 1992).
18. U. Trottenberg, C. Oosterlee, and A. Schüller, *Multigrid* (Academic Press, 2000).