Fully implicit nonstationary flow simulations with a monolithic off-lattice Boltzmann approach

T. Hübner, R. Mahmood, S. Turek
Institute for Applied Mathematics, TU Dortmund, Germany

Abstract: In this paper[1], the previously described monolithic approach [6] for the stationary discrete Boltzmann equation is extended to time-dependent problems. In general, both collision and advection operators are discretized on nonuniform grids as opposed to the standard Lattice Boltzmann method. Implicit time-stepping schemes are applied for an accurate and robust numerical treatment of the nonstationary flow problems. The resulting coupled system of equations is treated using special numerical methods for PDE's. As in the steady case, we apply a full Newton method for the nonlinear problems, but we also discuss possible variants of semi-implicit schemes all of which lead to nonsymmetric linear systems. The preconditioning of the used Krylov-space methods, resp., the construction of corresponding smoothing operators in the applied multigrid approaches is closely connected to the underlying short characteristic-upwinding discretization, yielding the exact inverse of the transport operators even for unstructured meshes due to a special numbering technique. Numerical results are given for the proposed solvers analysing the efficiency depending on the Mach number, time step and mesh size, while accuracy and stability of the complete space-time discretization are demonstrated for prototypical flow configurations at various timesteps.

AMS Subject Classifications: 35A25, 65M06, 76D05, 76P05
Key words: Discrete Boltzmann equation, upwind discretization, implicit time-stepping, Krylov-methods, multigrid, Newton

1 Introduction

We consider the discrete Boltzmann equation with the nine-velocity BGK model [2] in 2D

$$\frac{\partial f_i}{\partial t} + \boldsymbol{\xi}_i \cdot \nabla f_i = -\frac{1}{\tau}(f_i - f_i^{eq}) \qquad i = 0, \dots, 8. \tag{1}$$

The usual lattice set is applied for the microscopic velocities $\boldsymbol{\xi}_i = c \cdot \mathbf{e}_i$ with the particle speed c and

$$\mathbf{e}_i \in \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix} \right\}.$$

In the BGK collision model, the distributions $f_i = f(t, \boldsymbol{x}, \boldsymbol{\xi}_i)$ are relaxed with a typical single rate τ towards an equilibrium, which is given for the incompressible variant (see [5]) by

$$f_i^{eq}(\rho, \mathbf{u}) = W_i \left[\rho + \rho_0 \left(\frac{3}{c^2}(\boldsymbol{\xi}_i \cdot \mathbf{u}) + \frac{9}{2c^4}(\boldsymbol{\xi}_i \cdot \mathbf{u})^2 - \frac{3}{2c^2}\mathbf{u}^2 \right) \right] \tag{2}$$

with $\rho_0 = 1$ and corresponding weights W_i. By summation one obtains macroscopic moments, first of all the density $\rho = \sum_i f_i$ and the momentum $\rho_0 \mathbf{u} = \sum_i \boldsymbol{\xi}_i f_i$. However, instead of deriving the D2Q9 Lattice Boltzmann equation from above discrete velocity model (DVM), we discretize (1) independently of a structured mesh. For the advection we use a finite difference characteristic-upwinding of second order accuracy. This discretization is applicable to arbitrary (triangular) grids and also other sets of discrete velocities, like the reduced D2Q7 model as shown in [7].

[1]This work is kindly supported by the BMBF (SKALB project, grant 01IH08003D) and the Higher Education Commission (HEC) of Pakistan.

As shown in [6], the main idea is to solve generalized quasi-stationary problems directly by using a fully implicit discretization of the equation $\boldsymbol{\xi}_i \cdot \nabla f_i + \frac{1}{\tau}(f_i - f_i^{eq}) = g_i$. For this monolithic approach we use the Newton scheme as nonlinear solver and linear iterative Krylov-space methods in combination with multigrid. Special preconditioning techniques and a *generalized equilibrium formulation* (GEF, see [6]) of the discrete algebraic system based on a direct transport solver yiel good convergence rates, especially using multigrid as preconditioner. Here, we extend this monolithic approach from steady to time-dependent flow problems.
First of all, Section 2 deals with the implicit collision/advection discretization for the time-dependent DVM. We introduce the algebraic system resulting from our off-lattice space discretization and discuss the related Mach number (Ma) influence beside the usual compressibility error. In Section 3 we describe the actual block-system appearing in nonlinear iterations in every time-step and continue by reformulating the system of equations with the GEF approach for an efficient treatment of the linear subproblems. Finally, in Section 4 we provide numerical tests analysing the influence of Δt, h and Ma onto the condition number of the system and discuss efficient solvers with special preconditioning. The accuracy of the (semi-)implicit time-stepping schemes is verified by the *Flow around Cylinder* benchmark [13], with a special focus on stable simulations using large Δt.

2 Decoupled discretization

A discretization without explicit coupling of the temporal and spatial grid was introduced in our monolithic approach in [6], which will be the basis for the following methodology.

Time discretization

We consider the nonstationary case in (1) and present a straightforward time-discretization. A numerical off-lattice treatment usually begins with integrating the equation over the interval $[t_n, t_{n+1}]$. With the usual collision term $\Omega_i = -\frac{1}{\tau}(f_i - f_i^{eq})$ we write accordingly

$$f_i^{n+1} - f_i^n + \Delta t \int_{t_n}^{t_{n+1}} \boldsymbol{\xi}_i \cdot \nabla f_i dt = \int_{t_n}^{t_{n+1}} \Omega_i dt.$$

In the following, as in [8], both integrals are evaluated by a one-step θ-scheme,

$$f_i^{n+1} - f_i^n + \Delta t[\theta \boldsymbol{\xi}_i \cdot \nabla f_i^{n+1} + (1-\theta)\boldsymbol{\xi}_i \cdot \nabla f_i^n] = \Delta t[\theta \Omega_i^{n+1} + (1-\theta)\Omega_i^n] \quad \theta \in [0,1], \tag{3}$$

so that for nonzero θ the advection is treated implicitly, while for example in [4] and [9] it is evaluated explicitly at time t^n which leads to the well-known CFL restriction. Above semi-discrete equation yields for $\theta = 1$ the implicit Euler scheme

$$f_i^{n+1} + \Delta t \left(\boldsymbol{\xi}_i \cdot \nabla f_i^{n+1} - \Omega_i^{n+1}\right) = f_i^n,$$

but mostly $\theta = 1/2$ is chosen to obtain second order accuracy in the Crank-Nicolson scheme

$$f_i^{n+1} + \frac{\Delta t}{2}\left(\boldsymbol{\xi}_i \cdot \nabla f_i^{n+1} - \Omega_i^{n+1}\right) = \frac{\Delta t}{2}\left(\Omega_i^n - \boldsymbol{\xi}_i \cdot \nabla f_i^n\right) + f_i^n. \tag{4}$$

Space discretization

The finite difference characteristic-upwinding of second order as presented in [6] yields lower triangular matrices for each transport direction due to a topological resorting of the unknowns. Let $\nabla_{h,i}$ be an approximation of order γ of the characteristic derivative in the i-th direction. Then, the discretization scheme as mentioned above applied to the transport term in Eq. (1) yields

$$\boldsymbol{\xi}_i \cdot \nabla f_i = c(\boldsymbol{e}_i \cdot \nabla f_i) = c(\nabla_{h,i} f_i + O(h^\gamma)) = c\nabla_{h,i} f_i + O(c\ h^\gamma) \tag{5}$$

for unity lattice vectors $\boldsymbol{e}_i$. As we can see, the error due to the finite grid spacing is amplified by the sound speed parameter c. With the relation $Ma = O(1/c)$ we have therefore a specific discretization error $O(\mathrm{Ma}^{-1}h^\gamma)$ with an inverse Mach number influence which is opposed to the compressibility error $O(\mathrm{Ma}^2)$ in approximating the incompressible Navier-Stokes equations (see [12]). A balancing of the two contributions is achieved by setting the simulation parameters according to

$$h^\gamma = O(\mathrm{Ma}^3) = O(1/c^3)$$

which then yields the optimum asymptotic and quadratic convergence in the Mach number (for example, halving the grid spacing h during grid refinement, increase c by a factor of $2^{2/3}$ when using second order upwinding). The assumption was confirmed in [6] by a numerical analysis of the L_2-error for different steady-state CFD problems.
Now, in order to write the algebraic system and in view of an upcoming reformulation we identify two operators. First, we sum up the linear discrete transport and identity terms from Eq. (1) and denote by

$$\mathbf{T}_i f_i := f_i + \theta\Delta t[c\nabla_{h,i} f_i + \frac{1}{\tau} f_i].$$

Second, we take the remaining part of the collision term which is the equilibrium operator

$$\sum_k w_{ik} f_k := f_i^{eq}(\rho, \mathbf{u}) = W_i \left[\rho + \rho_0 \left(\frac{3}{c^2}(\boldsymbol{\xi}_i \cdot \mathbf{u}) + \frac{9}{2c^4}(\boldsymbol{\xi}_i \cdot \mathbf{u})^2 - \frac{3}{2c^2}\mathbf{u}^2\right)\right]. \tag{6}$$

In short, we have the nonlinear form $f_i^{eq} = \sum_k w_{ik} f_k$ where $w_{ik} = w_{ik}(c, \mathbf{u})$ are the corresponding weights obtained after resolving the equilibrium (2) in terms of the distributions f_i. The discrete equation obtained for (3) so far in these terms is

$$\mathbf{T}_i f_i^{n+1} - \frac{\theta\Delta t}{\tau} \sum_k w_{ik} f_k^{n+1} = g_i^n \tag{7}$$

where the right hand side is taken from the previous time step

$$g_i^n = f_i^n - (1-\theta)\Delta t \left(c\nabla_{h,i} f_i^n + \frac{1}{\tau} f_i^n - \frac{1}{\tau} f_i^{n,eq}\right).$$

Finally, we can write the resulting block-system as

$$\left(\begin{bmatrix} \mathbf{T}_0 & & & \\ & \mathbf{T}_1 & & \\ & & \ddots & \\ & & & \mathbf{T}_8 \end{bmatrix} - \frac{\theta\Delta t}{\tau} \begin{bmatrix} w_{00} & w_{01} & \dots & w_{08} \\ w_{10} & w_{11} & & \vdots \\ \vdots & & \ddots & \vdots \\ w_{80} & \dots & \dots & w_{88} \end{bmatrix}\right) \begin{bmatrix} f_0^{n+1} \\ f_1^{n+1} \\ \vdots \\ f_8^{n+1} \end{bmatrix} = \begin{bmatrix} g_0^n \\ g_1^n \\ \vdots \\ g_8^n \end{bmatrix} \tag{8}$$

and have to deal with a coupled algebraic system of equations which consists of a linear operator on the left and a nonlinear operator on the right. For the former we exploit the favourable triangular matrix property per each block $\mathbf{T}_i$ due to a special renumbering of the unknowns in the finite difference upwinding scheme (see [6]). The latter is introduced by the equilibrium term f_i^{eq} and is purely local but couples all directions.

3 Efficient treatment of the algebraic systems

An implicit treatment of the advection in (1) on unstructured grids in order to overcome the CFL-restriction in general requires the solution of an algebraic system of type (8). Usually, one tries to avoid implicit numerical treatment of collisions. However, this is possible only on structured meshes — corresponding approaches were presented by Guo et al in [4] or in [1] using characteristic based transformations. In a general framework, a computationally efficient (explicit) treatment is often in contrast to stability. For example, extrapolation (in time) of the equilibrium $f_i^{eq,n+1} := 2f_i^{eq,n} - f_i^{eq,n-1}$ as in [9] or [8] is consistent but might considerably affect stability. Therefore, we analyse a monolithic approach and introduce efficient methods to solve the nonlinear problem followed by a new approach for the linear sub-problems.

Nonlinear treatment

Due to the equilibrium in the system (8) given by Eq. (6) and *only* due to the terms $u_\alpha u_\beta$ therein, we obtain a quadratic nonlinearity $f_i f_j$ in the primary simulation variables. For the linearization, one of the components is substituted as $\tilde{f}_i$, which yields $\tilde{\mathbf{u}} = \sum_i \boldsymbol{\xi}_i \tilde{f}_i$ in the macroscopic momentum. Accordingly, we redefine the weights $\tilde{w}_{ik} = w_{ik}(c, \tilde{\mathbf{u}})$ and obtain the generalized (linearized) equilibrium $\tilde{f}_i^{eq}$ by

$$\sum_k \tilde{w}_{ik} f_k := \tilde{f}_i^{eq} = W_i \left[\rho + \rho_0 \left(\frac{3}{c^2}(\boldsymbol{\xi}_i \cdot \mathbf{u}) + \frac{9}{2c^4}(\boldsymbol{\xi}_i \cdot \mathbf{u})(\boldsymbol{\xi}_i \cdot \tilde{\mathbf{u}}) - \frac{3}{2c^2}\mathbf{u} \cdot \tilde{\mathbf{u}} \right) \right]. \tag{9}$$

At this point, a simple extrapolation $\tilde{f}_j := 2f_j^n - f_j^{n-1}$ from the previous time steps yields a semi-implicit method which preserves the second order time-accuracy of the Crank-Nicolson scheme (4). However, this approximation would give more or less the same stability as extrapolation of the entire equilibrium term. A better approach in view of stability would be to iterate (8) until a fixed point is reached with a given tolerance. If we write the nonlinear system (8) more generally as

$$N(\mathbf{x})\mathbf{x} = \mathbf{b}$$

with $\mathbf{x}$ representing the complete solution vector including all distributions f_i and $\mathbf{b}$ the right hand side, then the nonlinearity can be solved by a simple (damped) fixed point iteration (with f_i^n as starting vector $\mathbf{x}^0$)

$$\mathbf{x}^{l+1} = \mathbf{x}^l - wN(\mathbf{x}^l)^{-1}(N(\mathbf{x}^l)\mathbf{x}^l - \mathbf{b}) \quad , \quad w > 0.$$

Much more efficient would be using the Newton method. To that purpose we write the system in residual form

$$R(\mathbf{x}) = N(\mathbf{x})\mathbf{x} - \mathbf{b} = 0.$$

The Jacobian $\left[\frac{\partial R(\mathbf{x}^l)}{\partial \mathbf{x}}\right]$ is then used in the following iterative scheme:

$$\mathbf{x}^{l+1} = \mathbf{x}^l - \left[\frac{\partial R(\mathbf{x}^l)}{\partial \mathbf{x}}\right]^{-1} R(\mathbf{x}^l) \tag{10}$$

In our case the Jacobian can be obtained analytically, the derivation cancels the constant terms in (8), preserves the linear terms $\mathbf{T}_i$ and only affects the equilibrium, resulting in the slightly different operator

$$\sum_k \bar{w}_{ik}^l f_k := \bar{f}_i^{eq} = W_i \left[\rho + \rho_0 \left(\frac{3}{c^2}(\boldsymbol{\xi}_i \cdot \mathbf{u}) + \frac{9}{c^4}(\boldsymbol{\xi}_i \cdot \mathbf{u})(\boldsymbol{\xi}_i \cdot \mathbf{u}^l) - \frac{3}{c^2}\mathbf{u} \cdot \mathbf{u}^l \right) \right].$$

It means that the coefficients $\bar{w}_{ik}^l$ have an extra factor of 2 in front of the quadratic terms compared to $\tilde{w}_{ik}$. In each step of algorithm (10) we solve $\left[\frac{\partial R(\mathbf{x}^l)}{\partial \mathbf{x}}\right] \Delta\mathbf{x} = R(\mathbf{x}^l)$ for the solution update Δx with the matrix:

$$\left[\frac{\partial R(\mathbf{x}^l)}{\partial \mathbf{x}}\right] = \left(\begin{bmatrix} \mathbf{T}_0 & & & \\ & \mathbf{T}_1 & & \\ & & \ddots & \\ & & & \mathbf{T}_8 \end{bmatrix} - \frac{\theta\Delta t}{\tau} \begin{bmatrix} \bar{w}_{00}^l & \bar{w}_{01}^l & \dots & \bar{w}_{08}^l \\ \bar{w}_{10}^l & \bar{w}_{11}^l & & \vdots \\ \vdots & & \ddots & \vdots \\ \bar{w}_{80}^l & \dots & \dots & \bar{w}_{88}^l \end{bmatrix} \right) \tag{11}$$

Generalized equilibrium formulation

For an efficient treatment of the resulting linear subproblems, we introduced in [6] the so-called *generalized equilibrium formulation* (GEF) for the monolithic steady approach. Now, as a starting point we take the general algebraic system from (7), now stripped from the temporal indices, linearized and rearranged to

$$\mathbf{T}_k f_k = \frac{\theta\Delta t}{\tau} \tilde{f}_k^{eq} + g_k \quad k = 0, ..., 8$$

The procedure is completely analogous for the linear sub-problems of the Newton system (11) with the terms $\bar{f}_i^{eq}$, resp., weights $\bar{w}_{ik}$. In both cases, we multiply (formally) by the inverse transport operator and obtain

$$f_k = \mathbf{T}_k^{-1}\left(\frac{\theta\Delta t}{\tau}\tilde{f}_k^{eq} + g_k\right) \quad k = 0, ..., 8.$$

Now, for all $i = 0, ..., 8$, we multiply with the corresponding weights $\tilde{w}_{ik}$ (or $\bar{w}_{ik}$) which yields:

$$\tilde{w}_{ik} f_k = \tilde{w}_{ik}\mathbf{T}_k^{-1}\left(\frac{\theta\Delta t}{\tau}\tilde{f}_k^{eq} + g_k\right) \quad k = 0, ..., 8$$

Summing up over k finally gives us an equation for the (generalized) equilibrium term:

$$\tilde{f}_i^{eq} = \sum_k \tilde{w}_{ik}\mathbf{T}_k^{-1}\frac{\theta\Delta t}{\tau}\tilde{f}_k^{eq} + \sum_k \tilde{w}_{ik}\mathbf{T}_k^{-1} g_k \quad i = 0, ..., 8$$

The GEF block-system now reads:

$$\left(I - \frac{\theta\Delta t}{\tau}\begin{bmatrix} \tilde{w}_{00}\mathbf{T}_0^{-1} & \tilde{w}_{01}\mathbf{T}_1^{-1} & \dots & \tilde{w}_{08}\mathbf{T}_8^{-1} \\ \tilde{w}_{10}\mathbf{T}_0^{-1} & \tilde{w}_{11}\mathbf{T}_1^{-1} & & \vdots \\ \vdots & & \ddots & \vdots \\ \tilde{w}_{80}\mathbf{T}_0^{-1} & \dots & \dots & \tilde{w}_{88}\mathbf{T}_8^{-1} \end{bmatrix}\right)\begin{bmatrix} \tilde{f}_0^{eq} \\ \tilde{f}_1^{eq} \\ \vdots \\ \tilde{f}_8^{eq} \end{bmatrix} = \begin{bmatrix} \sum_k \tilde{w}_{0k}\mathbf{T}_k^{-1} g_k \\ \sum_k \tilde{w}_{1k}\mathbf{T}_k^{-1} g_k \\ \vdots \\ \sum_k \tilde{w}_{8k}\mathbf{T}_k^{-1} g_k \end{bmatrix} \tag{12}$$

Obviously, the condition number of the above system is close to one for a small coefficient $\frac{\theta\Delta t}{\tau}$. From the identity $\frac{1}{\tau} = \frac{c^2}{3\nu}$ with the viscosity ν (see [12]), we follow that moderate Reynolds numbers and sound constants c ensure a transport dominated discrete Boltzmann equation which is favourable for the presented approach. At the same time, linear convergence rates in iterative schemes can be always improved by reducing Δt. Due to the lower triangular form of the $\mathbf{T}_i$, at least the diagonal entries of all $\mathbf{T}_i^{-1}$ are explicitly known [6]. Additional preconditioning of system (12) is still possible, and taking the inverse of the 9x9 local matrix-entries in a block-Jacobian approach gives a kind of collision preconditioner, denoted in the following by GEF(\). Even better results are expected from a multigrid preconditioned approach (see [6]). We summarize the whole procedure, now combining the algebraic reformulation (GEF) with the Newton iteration.

1) In every time-step, the nonlinear system for the distributions f_i^{n+1} reads

$$\mathbf{T}_i f_i^{n+1} - \frac{\theta\Delta t}{\tau}\sum_k w_{ik} f_k^{n+1} = g_i^n \quad i = 0, ..., 8.$$

2) Per time-step, perform a sequence of Newton iterations for $l = 0, 1, \dots$ until a given tolerance is reached with the resulting linear block-systems

$$\left[\frac{\partial R(\mathbf{x}^l)}{\partial \mathbf{x}}\right] = \left(\begin{bmatrix} \mathbf{T}_0 & & & \\ & \mathbf{T}_1 & & \\ & & \ddots & \\ & & & \mathbf{T}_8 \end{bmatrix} - \frac{\theta\Delta t}{\tau}\begin{bmatrix} \bar{w}_{00}^l & \bar{w}_{01}^l & \dots & \bar{w}_{08}^l \\ \bar{w}_{10}^l & \bar{w}_{11}^l & & \vdots \\ \vdots & & \ddots & \vdots \\ \bar{w}_{80}^l & \dots & \dots & \bar{w}_{88}^l \end{bmatrix}\right).$$

3) An improved condition number is obtained by including the inverse transport operator into the system with the GEF approach which allows additional collision-oriented preconditioning

$$\left[\frac{\partial R(\mathbf{x}^l)}{\partial \mathbf{x}}\right]^{GEF} = \left(\mathbf{I} - \frac{\theta\Delta t}{\tau}\begin{bmatrix} \bar{w}_{00}^l\mathbf{T}_0^{-1} & \bar{w}_{01}^l\mathbf{T}_1^{-1} & \cdots & \bar{w}_{08}^l\mathbf{T}_8^{-1} \\ \bar{w}_{10}^l\mathbf{T}_0^{-1} & \bar{w}_{11}^l\mathbf{T}_1^{-1} & & \vdots \\ \vdots & & \ddots & \vdots \\ \bar{w}_{80}^l\mathbf{T}_0^{-1} & \cdots & \cdots & \bar{w}_{88}^l\mathbf{T}_8^{-1} \end{bmatrix}\right).$$

4 Numerical Results

We consider two classical CFD benchmark problems, the *Driven Cavity* problem and the *Flow around Cylinder* benchmark as described in [13]. For the driven cavity problem with small up to moderate Reynolds numbers, the flow is not time dependent. However, we consider this case to analyse our basic solvers and the influence of time-stepping on the convergence of the linear, resp., nonlinear solvers.

4.1 Comparison of basic solvers for Driven Cavity

A comparison of the nonlinear solvers is given in Table 1 for the Driven Cavity configuration. We take the converged solution from the previous level by prolongation as starting guess and perform one fully nonlinear sweep of the stationary solver ($\Delta t = \infty$) and one with $\Delta t = 1$, observing that the two configurations give quite similar results. For the smaller Δt we perform a corresponding number of timesteps to reach 0.5 seconds, showing the averaged number of nonlinear iterations for one timestep. Despite the low Reynolds number of 100, we need quite many fixed point (FP) iterations for the monolithic steady approach, while the Newton results are very good. In a time-stepping (for example Implicit Euler), smaller steps typically around $\Delta = 0.01$ act as a damping of the nonlinearity, so that the fixed point scheme is improving significantly.

		$c = 1$		$c = 10$		$c = 100$	
Δt	Grid points	Newton	Fixed Point	Newton	Fixed Point	Newton	Fixed Point
∞	1089	4	18	3	11	3	6
	4225	3	20	3	17	3	10
	16641	3	22	3	21	4	16
	66049	3	23	4	22	5	26
1	1089	3	18	3	10	3	6
	4225	3	19	3	15	3	9
	16641	3	20	3	19	4	14
	66049	3	20	4	20	4	19
0.1	1089	2	10	3	7	3	5
	4225	2	11	3	8	3	6
	16641	2	11	3	10	3	8
	66049	2	11	3	11	3	10
0.01	1089	2	3	2	3	2	4
	4225	2	4	2	4	2	4
	16641	2	4	2	4	2	4
	66049	2	4	2	4	2	4
0.001	1089	1	2	1	2	1	2
	4225	1	2	1	2	1	2
	16641	1	2	1	2	1	2
	66049	1	2	1	2	1	2

Table 1: Driven Cavity $Re = 100$: Absolute ($\Delta t = \infty$ and 1) and averaged No. of nonlinear iterations per time step to reduce the nonlinear defect by 10^{-6}, using Implicit Euler

In [6], we computed directly the stationary limit of the flow problems. Preconditioning of the linear system by a coarse grid correction in each iteration was essential, especially for low Mach numbers (i.e. large values of c). We assume that the linear condition number will improve with smaller Δt. Therefore, we present at first the single grid rates by turning off the coarse grid correction (always 16 smoothing steps are performed as 1 macro iteration) and compare GEF to GEF(\) with collision preconditioning (see Table 2). For the case $c = 1$, the linear convergence is excellent and mostly level-independent, due to the transport inverse being implicitly included in the GEF system matrix. In general, $\Delta t = 0.01$ yields significant improvement compared to $\Delta t = 1$. However, the extreme case $c = 100$ means slow convergence of the single grid iteration. With additional preconditioning in GEF(\) the rates improve significantly for small refinement levels, but barely change on the highest level, in which case also $\Delta = 0.01$ yields rates

around 0.9. Consequently, GEF(\) is most suited as a coarse grid solver. Otherwise, the use of multigrid (gain 2 digits on the coarser mesh) should stabilize the linear convergence, while in the moderate cases $c = 1$ and 10, and using small time-steps, the multigrid overhead can be mostly spared. Therefore, we compare single and two-grid convergence in Fig. 1, using the same configuration as before and showing plots for the three highest levels and GEF(\) only. For the case $\Delta t = 0.01$ on the right, the two variants both converge with few iterations, except for $c = 100$, where we obtain good results with coarse grid correction, but observe strong level-dependence of the single grid curves. For a large time step of $\Delta t = 1$ on the other hand (comparable to the monolithic steady approach), multigrid shows always significant improvements, with the exception of the case $c = 1$, which is by construction well suited for the GEF approach.

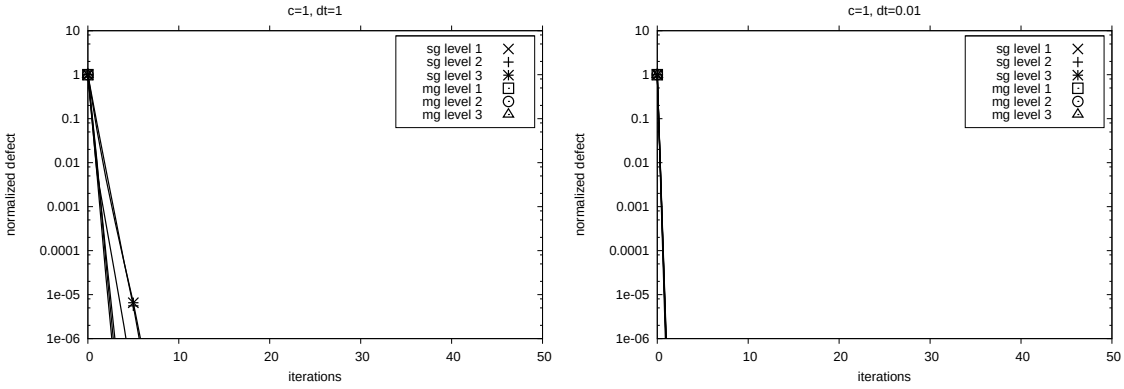
		$c = 1$		$c = 10$		$c = 100$	
Δt	Grid points	GEF	GEF(\)	GEF	GEF(\)	GEF	GEF(\)
1	1089	8.35E-2	2.55E-2	7.44E-1	5.89E-1	9.27E-1	7.67E-1
	4225	1.40E-1	6.18E-2	8.35E-1	7.58E-1	9.65E-1	8.94E-1
	16 641	1.45E-1	8.74E-2	8.75E-1	8.47E-1	9.91E-1	9.73E-1
	66 049	1.59E-1	1.02E-1	8.95E-1	8.90E-1	9.90E-1	9.81E-1
0.01	1089	6.17E-7	9.46E-7	2.32E-2	1.96E-3	7.56E-1	3.58E-1
	4225	1.89E-7	4.91E-7	7.31E-2	2.34E-2	8.41E-1	5.97E-1
	16 641	2.48E-7	9.36E-7	1.24E-1	8.60E-2	9.00E-1	7.74E-1
	66 049	2.30E-7	1.89E-7	1.59E-1	1.50E-1	9.22E-1	8.47E-1

Table 2: Driven Cavity $Re = 100$: Linear single grid convergence rates using 16 GMRES steps each

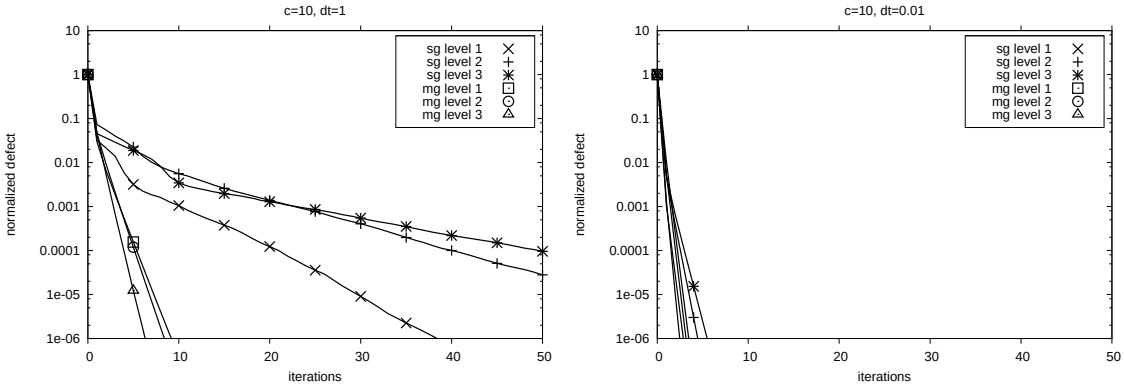
We extend the tests to the case Re=5000, too, where the optimal parameter range is known to be around $c = 10$ (compare [6]). Regarding the averaged nonlinear iterations per timestep in Table 3 with the Re=100 solution as starting guess, for large Δt and number of unknowns, the nonlinear convergence can fail and then FP requires explicit damping (here $w = 0.75$), while the Newton method performs remarkably well. Additionally, it is shown that the linear solver can deal with the high Reynolds number by use of multigrid and that the GEF(\) has a stronger impact on the convergence rates. However, on the highest refinement level, using multigrid with plain GEF smoothing steps is quite similar to smoothing with GEF(\), which is also visible in the linear convergence plots in Fig. 2. Again, we observe that collision preconditioning takes stronger effect on small levels and otherwise may be omitted to optimize the runtime.

		nonlinear		linear single grid		linear two-grid	
Δt	Grid points	Newton	Fixed Point	GEF	GEF(\)	GEF	GEF(\)
1	1089	3	15	9.51E-1	7.24E-1	5.67E-1	4.27E-1
	4225	4	21	9.84E-1	9.19E-1	6.13E-1	5.45E-1
	16641	5	25	9.96E-1	9.76E-1	6.14E-1	5.95E-1
	66049	5	32	9.97E-1	9.91E-1	6.64E-1	7.55E-1
0.01	1089	3	7	2.23E-1	1.29E-4	1.48E-1	1.15E-4
	4225	3	8	5.91E-1	9.48E-2	2.35E-1	3.76E-2
	16641	4	9	7.26E-1	2.82E-1	2.67E-1	1.05E-1
	66049	4	10	8.20E-1	5.41E-1	3.32E-1	2.38E-1

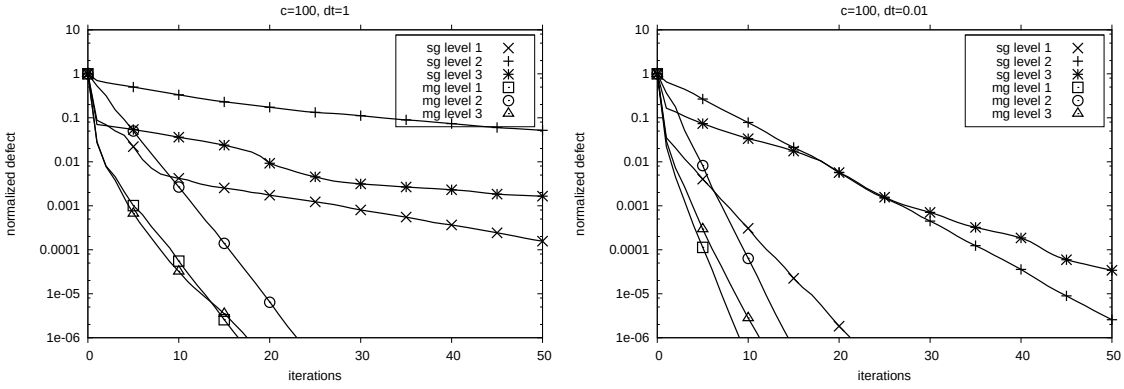
Table 3: Driven Cavity $Re = 5000$: Results of nonlinear/linear solvers to gain 6 digits, case c=10, nonlinear results with $Re = 100$ as starting guess



(a) Convergence for $c = 1$

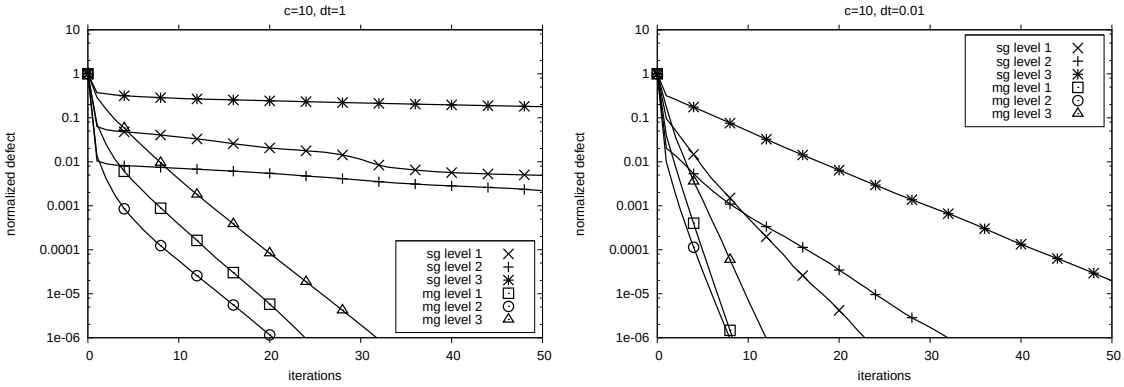


(b) Convergence for $c = 10$

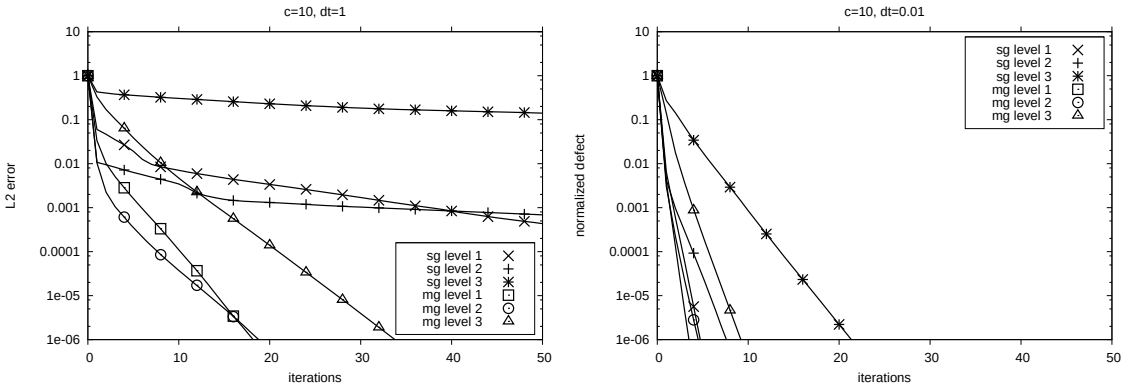


(c) Convergence for $c = 100$

Figure 1: Driven Cavity $Re = 100$: Convergence of linear defect using GMRES macro steps with GEF(\) (single-grid and MG preconditioned), 4225 up to 66049 grid points, $\Delta t = 1$ vs. $\Delta t = 0.01$



(a) Convergence for $c = 10$, GEF(plain)



(b) Convergence for $c = 10$, GEF(\)

Figure 2: Driven Cavity $Re = 5000$: Convergence of linear defect using GMRES macro steps (single-grid and MG preconditioned), 4225 up to 66049 grid points, $\Delta t = 1$ vs. $\Delta t = 0.01$

4.2 Numerical results for the Flow around Cylinder benchmark

The *Flow around Cylinder* benchmark [13] is used here to test the accuracy of the applied time-discretization schemes and the space-discretization on locally adapted unstructured grids (see Fig. 3). At first, taking a viscosity of $\nu = 0.001$ and parabolic inflow profile with maximum value $u_{max} = 0.3$ results in a Reynolds number of 20 and time-independent flow (as in [6]). This can be solved with our monolithic steady approach, or with a pseudo time-stepping and arbitrary Δt. As seen in Table 4, the MG preconditioned solver shows level-independent convergence rates and significant advantages when large timesteps are performed which vastly influences the condition number of the linear system.

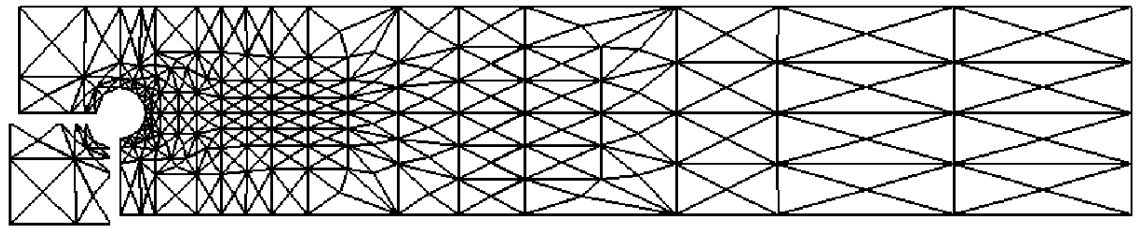
4k grid	single grid		twogrid	
	GEF	GEF(\)	GEF	GEF(\)
$\Delta t = 1$	9.96E-1	9.69E-1	8.44E-1	6.93E-1
$\Delta t = 0.01$	7.76E-1	6.03E-1	5.82E-1	3.62E-1
$\Delta t = 0.0001$	1.36E-4	9.74E-7	4.04E-5	3.44E-7
16k grid	single grid		twogrid	
	GEF	GEF(\)	GEF	GEF(\)
$\Delta t = 1$	9.95E-1	9.88E-1	8.44E-1	6.75E-1
$\Delta t = 0.01$	8.22E-1	7.07E-1	5.93E-1	3.48E-1
$\Delta t = 0.0001$	2.16E-4	6.41E-7	6.38E-5	1.21E-7
66k grid	single grid		twogrid	
	GEF	GEF(\)	GEF	GEF(\)
$\Delta t = 1$	9.95E-1	1.00E+0	8.65E-1	7.36E-1
$\Delta t = 0.01$	8.34E-1	8.09E-1	6.09E-1	4.07E-1
$\Delta t = 0.0001$	1.95E-4	1.77E-5	5.74E-5	1.36E-5

Table 4: *Flow around Cylinder* Re=20: Linear single grid and MG convergence rates for different mesh levels and time steps, c=10

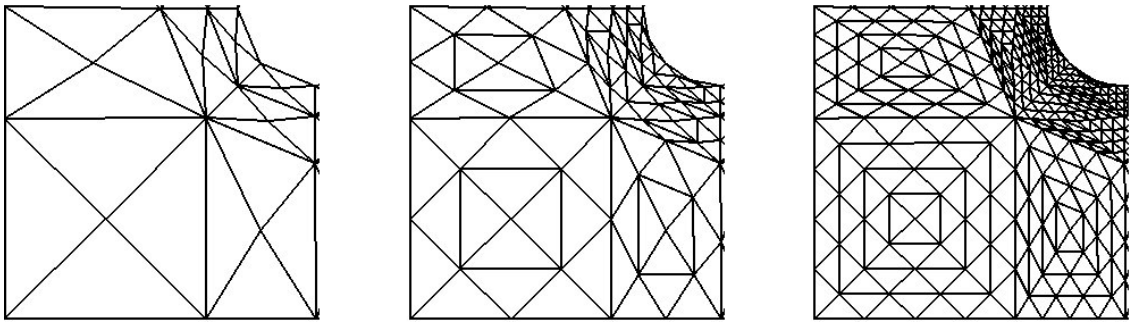
Next, the inflow velocity is increased to $u_{max} = 1.5$ which results in $Re = 100$ and a transient flow regime. Now the time-step has to be reduced to simulate the periodic flow behaviour, at least to obtain accurate results for the period time of $T_{period} \sim 0.33$. With the use of multigrid one obtains a robust and efficient linear solver for quite large Δt (see Table 5), while for the semi-implicit approach with a much smaller range of stable Δt (for example $\Delta t \sim 10^{-3}$, see Table 9) the single grid solver is sufficient.

4k grid	single grid		twogrid	
	GEF	GEF(\)	GEF	GEF(\)
$\Delta t = 0.01$	6.06E-1	3.76E-1	4.22E-1	2.29E-1
$\Delta t = 0.001$	2.83E-2	1.63E-3	1.80E-2	1.40E-3
$\Delta t = 0.0001$	5.88E-7	2.44E-7	3.66E-8	1.95E-8
16k grid	single grid		twogrid	
	GEF	GEF(\)	GEF	GEF(\)
$\Delta t = 0.01$	6.75E-1	5.35E-1	4.46E-1	2.55E-1
$\Delta t = 0.001$	4.76E-2	1.35E-2	2.49E-2	5.88E-3
$\Delta t = 0.0001$	8.07E-7	6.83E-7	4.16E-8	1.93E-8
66k grid	single grid		twogrid	
	GEF	GEF(\)	GEF	GEF(\)
$\Delta t = 0.01$	7.24E-1	6.95E-1	4.97E-1	3.18E-1
$\Delta t = 0.001$	6.80E-2	4.66E-2	2.63E-2	1.70E-2
$\Delta t = 0.0001$	5.33E-7	6.00E-7	8.68E-9	4.26E-9

Table 5: *Flow around Cylinder* Re=100: Linear single grid and MG convergence rates, c=10



(a) Channel



(b) Grid hierarchy

Figure 3: Unstructured grid for *Flow around Cylinder*

To analyse the accuracy (in time and space) of the nonstationary flow simulations at $Re = 100$, we look at the maximum, resp., minimum drag and lift values during one period and the corresponding Strouhal number. The first order Implicit Euler scheme produces significant time-discretization errors as can be seen in Table 6. In comparison, the second order Crank-Nicolson scheme allows to use larger time steps, with results for $\Delta t = 1/100$ being quite similar to $\Delta t = 1/10000$ (see Tables 7 and 8). The proposed semi-implicit approach gives also second order accuracy in time, but with reduced stability which is dependent on the refinement level as seen in Table 9. The semi-implicit results in Table 10 are similar to the Crank-Nicolson numbers, but needed up to 40 times the number of timesteps of the stable, fully-implicit approach.

	4K grid			16K grid		
c	C_{Lmax}	C_{Dmax}	St	C_{Lmax}	C_{Dmax}	St
	$\Delta t = 1/10000$			$\Delta t = 1/10000$		
2	0.67	2.89	0.1991	0.84	3.21	0.2152
4	1.09	3.17	0.2515	0.90	3.36	0.2670
8	0.85	3.00	0.2777	1.09	3.25	0.2917
16	0.67	2.93	0.2761	0.92	3.17	0.3012
	$\Delta t = 1/1000$			$\Delta t = 1/1000$		
2	0.63	2.89	0.1961	0.83	3.18	0.2128
4	1.06	3.14	0.2500	0.89	3.33	0.2632
8	0.80	2.99	0.2778	1.04	3.21	0.2857
16	0.62	2.92	0.2703	0.85	3.14	0.2941
	$\Delta t = 1/100$			$\Delta t = 1/100$		
2	0.42	2.80	0.1887	0.57	3.03	0.2041
4	0.63	2.92	0.2381	0.55	3.08	0.2500
8	0.37	2.86	0.2564	0.42	3.01	0.2778
16	0.15	2.84	0.2632	0.29	2.99	0.2941

Table 6: Benchmark results: Implicit Euler, references: $C_{Lmax} = 0.987$, $C_{Dmax} = 3.230$, $St = 0.300$

Finally, we present a more elaborate version of the *Flow around Cylinder* benchmark, using the time-dependent inflow profile $u(t,x) = u(x) \cdot sin(t\frac{\pi}{8})$ (with $u(x)$ as previously) and analyse the forces acting on the cylinder for the whole 8 seconds of simulation time (see again [13] and *www.featflow.de*). In Figure 4 we obtain excellent results for the pressure difference Δp between front and back of the cylinder, when we use extrapolation of the pressure on the boundary. However, as described in [7] this has no influence on the drag and lift when using the force evaluation method proposed in [10]. With grid refinement, we increase successively c according to the asymptic behaviour mentioned earlier and obtain a good approximation of the reference drag. The lift shows larger oscillations and is more difficult to reproduce, as shown in Figure 5. However, we obtain the right period on the highest refinement level and a good approximation of the amplitude when the Mach number is sufficiently reduced. In the same figure, the accuracy using the quite large time step of $\Delta t = 1/100$ is compared to a semi-implicit simulation with $\Delta t = 1/4000$ due to Table 9. The advanced performance in time becomes abundantly clear from the contiguous graphs over the whole simulation time.
Nevertheless, the results are not perfect and in a next step we will try to increase the order of the spatial discretization. The Discontinuous Galerkin (DG) method (see [3] for an early example of DG applied to the discrete Boltzmann equation) is a possible extension to our method which allows to preserve the developed upwinding techniques, while basis functions of arbitrary order can be used, especially in crucial parts of the geometry.

	4K gridpoints			16K gridpoints		
c	C_{Lmax}	C_{Dmax}	St	C_{Lmax}	C_{Dmax}	St
	$\Delta t = 1/10000$			$\Delta t = 1/10000$		
2	0.67	2.89	0.1992	0.84	3.21	0.2155
4	1.10	3.17	0.2516	0.90	3.37	0.2672
8	0.85	3.01	0.2779	1.10	3.26	0.2920
16	0.68	2.94	0.2762	0.93	3.18	0.3014
	$\Delta t = 1/1000$			$\Delta t = 1/1000$		
2	0.67	2.89	0.2000	0.85	3.22	0.2151
4	1.11	3.19	0.2519	0.91	3.37	0.2674
8	0.85	3.01	0.2786	1.11	3.25	0.2915
16	0.68	2.93	0.2762	0.93	3.18	0.3012
	$\Delta t = 1/100$			$\Delta t = 1/100$		
2	0.66	2.90	0.2041	0.85	3.21	0.2174
4	1.08	3.19	0.2500	0.92	3.37	0.2632
8	0.84	3.01	0.2778	1.10	3.24	0.2941
16	0.68	2.93	0.2703	0.90	3.16	0.3030

Table 7: Benchmark results: Crank-Nicolson, references: $C_{Lmax} = 0.987$, $C_{Dmax} = 3.230$, $St = 0.300$

66K gridpoints					
c	C_{Lmin}	C_{Lmax}	C_{Dmin}	C_{Dmax}	St
$\Delta t = 1/1000$					
4	-0.925	0.908	3.082	3.373	0.2710
8	-1.211	1.165	3.163	3.278	0.2924
12	-1.117	1.081	3.138	3.228	0.2985
16	-1.054	1.019	3.119	3.202	0.3003
20	-1.027	0.992	3.110	3.189	0.3021
24	-0.998	0.968	3.104	3.170	0.3021
$\Delta t = 1/200$					
16	-1.055	1.022	3.119	3.203	0.3030
20	-1.029	0.994	3.111	3.188	0.3030
24	-1.004	0.973	3.105	3.170	0.3030
$\Delta t = 1/100$					
16	-1.054	1.016	3.117	3.200	0.3030
20	-1.028	0.994	3.109	3.187	0.3030
24	-1.010	0.979	3.106	3.171	0.3030

Table 8: Benchmark results: Crank-Nicolson, high refinement level, references: $C_{Lmin} = -1.023$, $C_{Lmax} = 0.987$, $C_{Dmin} = 3.168$, $C_{Dmax} = 3.230$, $St = 0.300$

grid points	4 264	16 848	66 976	267 072
stable Δt	1/500	1/1000	1/2000	1/4000

Table 9: Stability limits for semi-implicit simulation of the benchmark at Re=100

c	C_{Lmin}	C_{Lmax}	C_{Dmin}	C_{Dmax}	St
16k grid, $\Delta t = 1/1000$					
8	-1.166	1.103	3.142	3.254	0.2915
12	-1.049	1.005	3.125	3.194	0.2994
16	-0.967	0.925	3.112	3.175	0.3021
20	-0.921	0.877	3.112	3.163	0.3021
66k grid, $\Delta t = 1/2000$					
12	-1.115	1.078	3.139	3.229	0.2981
16	-1.055	1.020	3.119	3.202	0.3008
20	-1.031	0.998	3.111	3.185	0.3017
24	-1.004	0.973	3.111	3.164	0.3035
267k grid, $\Delta t = 1/4000$					
12	-1.110	1.076	3.118	3.274	0.2974
16	-1.063	1.026	3.175	3.223	0.3008
20	-1.045	1.009	3.126	3.237	0.3005
24	-1.030	1.004	3.122	3.235	0.3010

Table 10: Benchmark results: second order semi-implicit, references: $C_{Lmin} = -1.023$, $C_{Lmax} = 0.987$, $C_{Dmin} = 3.168$, $C_{Dmax} = 3.230$, $St = 0.300$

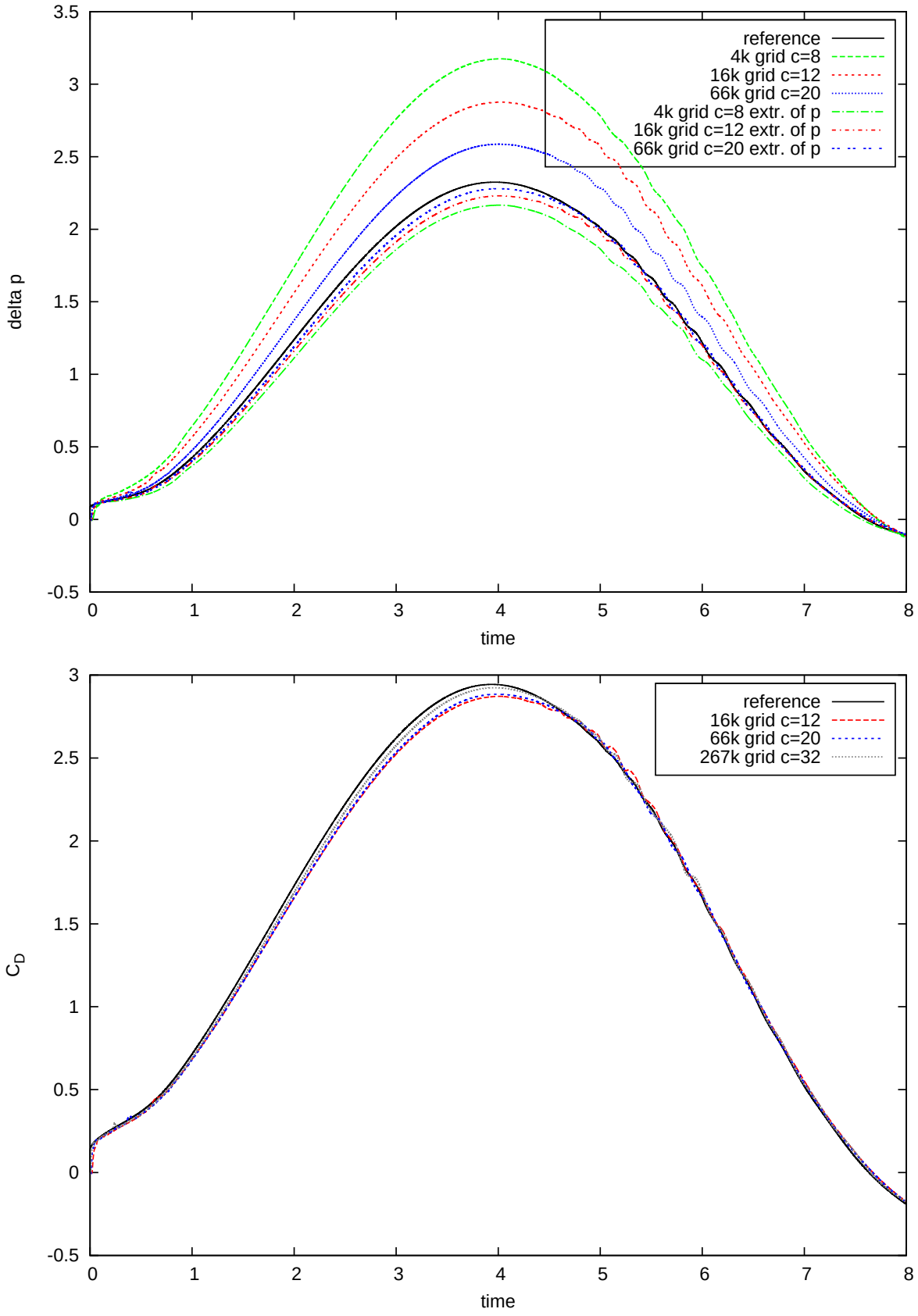


Figure 4: *Flow around Cylinder* with 8 seconds inflow pulse: Δp and drag

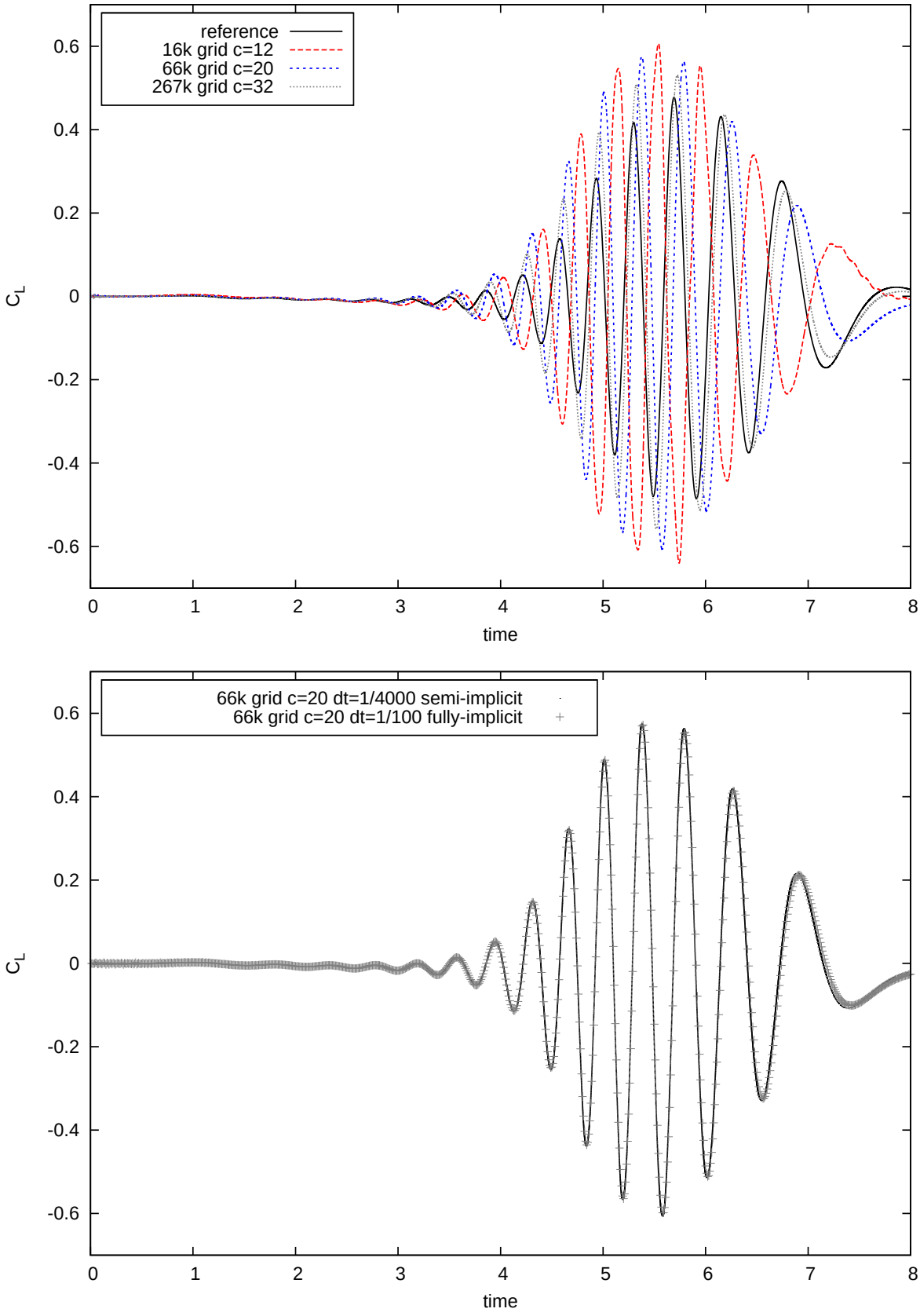


Figure 5: *Flow around Cylinder* with 8 seconds inflow pulse: lift

5 Conclusion

We have shown that the methodology introduced in [6] is applicable to time-dependent flow problems on unstructured meshes. The fully implicit discretization of the discrete Boltzmann equation ensures stability beyond micro-timesteps, but (at least) second order schemes in space and in time are required for accurate simulations of the presented benchmarks. As in the monolithic steady approach, the Newton method is used to solve efficiently the nonlinear systems. The condition number of the resulting linear subproblems is again improved by an algebraic reformulation via the inverse transport. Moreover, this GEF approach allows for additional preconditioning by coarse grid corrections and we obtain thus a robust multigrid solver also in the caseof low Mach number and large Δt.
Regarding future developments, a promising way to combine the upwinding approach with finite element methods can be realized using Discontinuous Galerkin (DG) schemes. Similar to [11], an efficient ordering is then applied elementwise and again yields a direct transport solver. Corresponding research is ongoing and we expect advantages using higher order DG approximations especially compared to the Finite Difference upwinding which naturally has to use first order differences at the boundary.

References

[1] Bardow, A., Karlin, I.V., Gusev, A.A., *General characteristic-based algorithm for off-lattice Boltzmann simulations*, Europhys. Letters **75**, No. 3, 434–440 (2006)

[2] Bhatnagar, P.L., Gross, P.L., Krook, M., *A model for collision processes in gases*, Physical Review **94**, 511 (1954)

[3] Düster, A., Demkowicz, L., Rank, E., *High-order finite elements applied to the discrete Boltzmann equation*, Int. J. of Num. Meth. in Eng. **67**, 1094–1121 (2006)

[4] Guo, Z., Zhao, T. S., *Explicit finite-difference lattice Boltzmann method for curvilinear coordinates*, Phys. Rev. E **67**, 066709 (2003)

[5] He, X., Luo, L. S., *Lattice Boltzmann model for the incompressible Navier-Stokes equation*, J. of Stat. Phys. **88**, 927–944 (1997)

[6] Hübner, T., Turek, S., *Efficient monolithic simulation techniques for the stationary Lattice Boltzmann equation on general meshes*, Computing and Visualization in Science **13**, No. 3, 129–143 (2010)

[7] Hübner, T., *A monolithic, off-lattice approach to the discrete Boltzmann equation with fast and accurate numerical methods*, PhD Thesis, TU Dortmund (2010), `http://hdl.handle.net/2003/27715`

[8] Li, Y., LeBoeuf, E., Basu, P.K., *Least-squares finite-element scheme for the lattice Boltzmann method on an unstructured mesh*, Phys. Rev. E **72**, 046711 (2005)

[9] Mei, R., Shyy, W., *On the finite difference-based lattice Boltzmann method in curvilinear Coordiates*, J. of Comp. Phys. **143**, 426–448 (1998)

[10] Mei, R., Yu, D., Shyy, W. and Luo, L.-S., *Force evaluation in the Lattice Boltzmann method involving curved geometry*, Physical Review E, **65**, 041203 (2002)

[11] Natvig J. R.,Lie, K.-A., Eikemo, B., Berre, I.: An efficient discontinuous Galerkin method for advective transport in porous media, Advances in Water Resources **30**, No. 12, 2424–2438 (2007)

[12] Reider, M., Sterling, J., *Accuracy of discrete velocity BGK models for the simulation of the incompressible Navier Stokes equations*, Computers and Fluids **24**, 459–467 (1995)

[13] Schäfer, M., Turek, S.: Benchmark computations of laminar flow around cylinder. Notes on Numerical Fluid Mechanics **52**, 547–566 (1996)