Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 224 (2007) 100–116
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

On the influence of different stabilisation methods for the incompressible Navier–Stokes equations ☆

Sandra Nägele, Gabriel Wittum *

Technische Simulation, IfI, Universität Heidelberg, Im Neuenheimer Feld 368, 69120 Heidelberg, Germany

Received 10 September 2006; received in revised form 22 December 2006; accepted 22 December 2006
Available online 19 January 2007

Abstract

Two different stabilisation methods for the incompressible Navier–Stokes equations are investigated. Both methods are based on a special interpolation scheme for the velocity components but with different effects and behaviour. They introduce varying errors in the continuity equation but also the derivation of terms contributing to the special interpolation have a great influence on the quality of the results. By means of various examples the two stabilisation methods and their contributing parts are compared.

Keywords: Stabilisation; Incompressible Navier–Stokes equations; Colocated scheme; Finite volume

1. Introduction

Flows of incompressible fluids are modelled by the Navier–Stokes equations

$$\frac{\partial \mathbf{u}}{\partial t} - \nabla \cdot (v(\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathrm{T}})) + \nabla \cdot (\mathbf{u}\mathbf{u}^{\mathrm{T}}) + \nabla p = 0$$
$$\operatorname{div}(\mathbf{u}) = 0$$

with viscosity v, velocity $\mathbf{u} = (u_1, \ldots, u)^{\mathrm{T}}$, d the spatial dimension and pressure p. For fluids subject to linear material laws, the viscosity v is constant. However, for small viscosities and large velocities the flow develops unordered small scale fluctuations of velocity and pressure. In this case the flow is called turbulent, but throughout this paper we will focus on the laminar flow regime only. The reason is, that we can focus on the influence of the stabilisation and the discretisation on the simulation results without having to take care of the model effects of a turbulence model. Some aspects of the turbulent case are considered in [11] but not thoroughly discussed concerning the stabilisation influence or contribution.

☆ This work was supported by the Deutsche Forschungsgemeinschaft, SFB 359 and SPP 1141.
* Corresponding author.
E-mail addresses: sandra.naegele@web.de (S. Nägele), wittum@iwr.uni-heidelberg.de (G. Wittum).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2006.12.024

We focus on unstructured or not necessarily structured grids, therefore we are interested in a flexible way to localise the unknowns. The most convenient way for this is to use a colocated arrangement of the variables. By this, one can use the same ansatz functions for the velocities and the pressure, but the resulting system is unstable and an unphysical checkerboard pressure distribution can result. Rhie and Chow [13] presented a pressure correction scheme and used special interpolations to solve this problem. There are other stabilisation methods like the one of Hughes et al. [8] based on a Galerkin/least squares formulation or the method described by Arnold et al. [2] which uses special finite elements to avoid the checkerboard problem. Schneider and Raw [17] introduced a very interesting stabilisation which we will describe in the following section. Some enhancements were introduced by Karimian and Schneider [9] for weakly compressible flows. But also for the incompressible case this modification has some advantages which will be addressed. Throughout this paper, we will focus on these two stabilisation techniques.

In the present paper we discuss the behaviour of the various stabilisation and discretisation possibilities and their effect on the quality of the solution. On the basis of several criteria these methods are investigated and the differences will be demonstrated experimentally since analytical methods are not available for nonlinear and/or complex flows. Although the discretisation was designed for complex flows, its influence on the quality of the solution is still an open question. Therefore, criteria are established to compare the discretisation methods before they are applied to complex flows.

The structure of the paper is as follows. First, the discretisation methods will be explained in Sections 2 and 3. In Section 4, the solution strategy will be briefly described. Finally, in Section 5, the criteria for comparison of the methods together with appropriate examples and their discussion will be presented.

2. Discretisation

The equations are discretised by a finite volume method based on a vertex centred scheme where all unknowns are located at the nodal or grid points. The control volumes are defined via dual boxes of the underlying finite element grid. A simple sketch of the resulting control volume in a 2d situation can be seen in Fig. 1. The construction, however, is general and applies to 3d as well.

After application of Gauss' theorem and splitting the integration over the whole control volume surface into a sum of integrations over subsurfaces, the resulting system in discretised form reads

$$\frac{\delta}{\delta t}|CV|U_i + \sum_{ip=1}^{\#ip(CV)} \left(u_i u_j n_j + p n_i - \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j \right) \Big|_{ip} = 0 \tag{1}$$

$$\sum_{ip=1}^{\#ip(CV)} (u_j n_j)|_{ip} = 0. \tag{2}$$

$\#ip(CV)$ denotes the number of subsurfaces of the control volume surface and is therefore equal to the number of integration points of the control volume CV. $|CV|$ denotes the area of the control volume in 2d and accordingly the volume in 3d. In Fig. 1, an example for a control volume is shown with 10 integration points or subsurfaces respectively. Small letters correspond to integration point quantities and have to be further

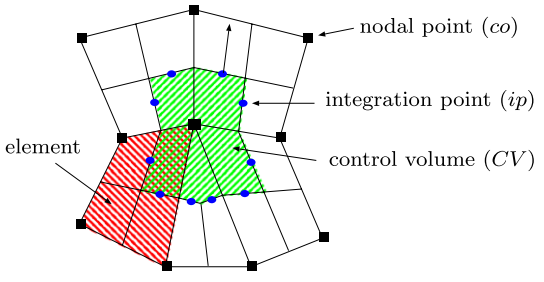

Fig. 1. Sketch of a 2d control volume, consisting of 5 subcontrol volumes (SCV).

specified whereas capital letters denote nodal quantities and can be evaluated directly. The outer normal of each subsurface $\mathbf{n} = (n_1, \ldots, n_d)^{\mathrm{T}}$ is scaled by the subsurface area to get a shorter notation. For the time discretisation $\frac{\delta}{\delta t}$ a diagonally implicit Runge–Kutta method of second order is applied, which is described in the paper of Alexander [1] for ordinary differential equations. If everything but the time derivative is summarised in the operator $K(\mathbf{u}, p, t)$

$$K(\mathbf{u}, p, t) = \begin{pmatrix} \sum_{ip=1}^{\#ip(CV)} \left(u_i u_j n_j + p n_i - \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j \right) \Big|_{ip} \\ \sum_{ip=1}^{\#ip(CV)} (u_j n_j) \Big|_{ip} \end{pmatrix}$$

while the operator T representing the time derivative looks like $T(\mathbf{u}, p, t) = \begin{pmatrix} |CV|\mathbf{u} \\ 0 \end{pmatrix}$, the time stepping scheme can be formulated in two steps:

$$T(\mathbf{u}, p, t + \alpha\Delta t) - T(\mathbf{u}, p, t) + \alpha\Delta t K(\mathbf{u}, p, t + \alpha\Delta t) = 0,$$
$$T(\mathbf{u}, p, t + \Delta t) - T(\mathbf{u}, p, t + \alpha\Delta t) + (1 - 2\alpha)\Delta t K(\mathbf{u}, p, t + \alpha\Delta t) + \alpha\Delta t K(\mathbf{u}, p, t + \Delta t) = 0,$$

where $\alpha = 1 - \frac{1}{2}\sqrt{2}$. This scheme is of order 2 and stable.

A quasi-Newton linearisation of the convection term yields

$$\sum_{j=1}^{d} u_i u_j n_j \approx \sum_{j=1}^{d} u_i \tilde{u}_j n_j, \tag{3}$$

where $\tilde{u}_j$ stands for the latest approximation of the integration point velocity u_j. The assembly of the discretised system can be done element-wise since mainly subsurface integrals have to be computed.

All unknowns are located in the nodes, thus the discretisation would be unstable if the integration point quantities are interpolated via the ansatz functions only, because the LBB-condition is not fulfilled in this case [6]. To stabilise the system, a special interpolation for the integration point velocities is constructed. The determination of this interpolation is based on the idea that the correct dependence of velocity and pressure is contained in the momentum equation itself. This kind of interpolation and stabilisation was developed by Schneider and Raw [17] and was further modified by Karimian and Schneider [9]. Both methods will be described in the following section. They are similar to the Galerkin/least squares technique of Hughes et al. [8], but the factors introduced are derived automatically instead of being predefined as in the least squares method.

3. Stabilisation and convection treatment

The stabilisation developed by Schneider and Raw [17] is called FIELDS (Finite Element Differential Scheme) and will be explained in Section 3.1. The stabilisation introduced by Karimian and Schneider [9] is designated for distinction by FLOW. The differences of FLOW in comparison to FIELDS will be addressed in Section 3.2. Section 3.3 focuses on the discretisation of the convection operator in the momentum equation.

3.1. FIELDS

The idea of FIELDS to derive a proper dependence of integration point velocities and the nodal pressure values is to derive a special interpolation for the velocities at each integration point. To determine this kind of interpolation for u_i in each element and in each integration point the momentum equation is approximated by a very simple finite difference approach, where the diffusion part is assumed to be a Laplacian for simplicity. The convection term is linearised and afterwards discretised by an upwind method

$$\sum_{j=1}^{d} u_j \frac{\partial u_i}{\partial x_j} \approx \sum_{j=1}^{d} \tilde{u}_j \frac{\partial u_i}{\partial x_j} = \|\tilde{\mathbf{u}}\| \frac{\partial u_i}{\partial s} \quad \text{where } s = \frac{1}{\|\tilde{\mathbf{u}}\|} \tilde{\mathbf{u}}. \tag{4}$$

Thus, the simplified form of the momentum equation is

$$\frac{\partial u_i}{\partial t} - \nu \Delta u_i + \|\tilde{\mathbf{u}}\| \frac{\partial u_i}{\partial s} + \frac{\partial p}{\partial x_i} = 0. \tag{5}$$

To explain the detailed form of the stabilisation, the position of all integration points as well as the local flow direction at integration point ip_4 are shown in Fig. 2.

Starting from Eq. (5) the finite difference approximation for one integration point reads

$$\frac{u_i - u_i^0}{\Delta t} - \nu \frac{\sum_{k=1}^{n_N} N_k(ip) U_i^k - u_i}{L_d^2} + \|\tilde{\mathbf{u}}\| \frac{u_i - u_i^{\mathrm{up}}}{L_c} + \sum_{k=1}^{n_N} \frac{\partial N_k}{\partial x_i} P_k = 0 \Bigg|_{ip_l}, \tag{6}$$

where u_i^0 denotes the integration point velocity at the previous time point $t - \Delta t$, L_d^2 is the diffusion length scale (the form of which will be derived shortly), U_i^k corresponds to the value of the ith velocity component in node k, u_i^{up} denotes the upwind velocity, L_c is the distance between the integration point ip and the corresponding upwind position up and $\tilde{\mathbf{u}}$ refers to the last approximation of the integration point velocity $\mathbf{u}$. N_k are the nodal ansatz functions with $k = 1, \ldots, n_N$ and n_N the number of nodes of the element.

This leads to a system of equations depending on integration point velocities, nodal velocities and nodal pressures which can be solved directly in each element. The resulting approximation for the integration point velocities

$$u_i = \left(\frac{1}{\Delta t} + \frac{\nu}{L_d^2} + \frac{\|\tilde{\mathbf{u}}\|}{L_c} \right)^{-1} \left(\frac{1}{\Delta t} u_i^0 + \frac{\nu}{L_d^2} \sum_{k=1}^{n_N} N_k U_i^k + \frac{\|\tilde{\mathbf{u}}\|}{L_c} u_i^{\mathrm{up}} - \sum_{k=1}^{n_N} \frac{\partial N_k}{\partial x_i} P^k \right)$$

are put into the continuity equation (2) to stabilise the system of equations. By doing so, a pressure dependence in form of a Laplacian scaled with a constant times the mesh size squared is introduced in the continuity equation, as known from other stabilised schemes.

3.1.1. Finite difference convection

Various upwinding methods can be applied to discretise the convection term (4). They differ in the choice of the upwind point *up* and therefore also in the length scale L_c and consequently lead to different errors. The simplest and most diffusive choice is the full upwind scheme illustrated in Fig. 3c. The upwind point for this example is $\mathbf{u}^{\mathrm{up}} = \mathbf{U}_3$ which is clearly a very bad approximation and far away from the streamline. Very similar to the full upwind method is the skewed upwind where not only the two nodes belonging to the nearest edge of the element are taken into account, but the closest upwind node of the element serves as approximation. This leads to the choice $\mathbf{u}^{\mathrm{up}} = \mathbf{U}_0$ as upwind point, which is an improvement but still very diffusive. A better approximation can be obtained by the Linear Profile Skewed method (LPS) displayed in Fig. 3a. This scheme determines the upwind point by intersection of the streamline and the element boundary. Very similar to the LPS upwind method is the regular upwind scheme described in [17], but throughout this paper we will only focus on the LPS scheme, since numerical experiments showed similar results. The resulting approximation for the upwind point in the example would be

$$\mathbf{u}^{\mathrm{up}} = \frac{b}{a+b} \mathbf{U}_0 + \frac{a}{a+b} \mathbf{U}_1. \tag{7}$$

This scheme has a higher approximation order than the other ones, but it can introduce negative coefficients into the matrix. To prevent this behaviour, Raw introduced a positive coefficient upwind method [16]

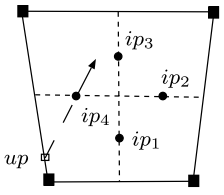

Fig. 2. Sketch of element with integration points.

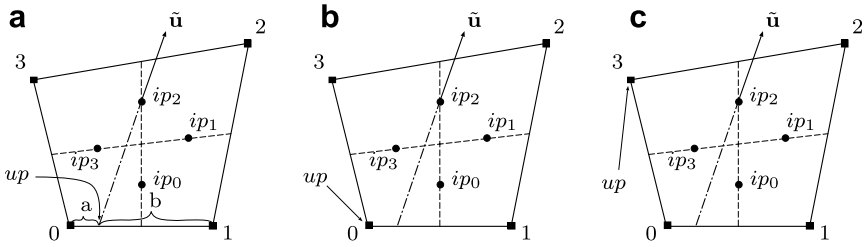

Fig. 3. Upwind methods: (a) LPS; (b) skewed upwind; (c) full upwind.

depending on the mass flow over the control volume faces. It does not introduce negative coefficients but it is more diffusive than the LPS scheme. For distinction between these two, it will be called POS throughout this paper. To describe the behaviour of the scheme, only a short description will be given, for more details we refer to [16] or [10]. Since we are looking for an upwind point we have $m_{ip} = \mathbf{u}(ip)\mathbf{n}(ip) > 0$. Thus, the amount of mass flowing out at this integration point ip must be flowing in through the node (k for instance) belonging to that subcontrol volume (SCV) or through the other faces of the subcontrol volume. The approximation of the velocity in the upwind point then reads

$$\mathbf{u}^{\mathrm{up}} = \left(1 - \frac{m_{\mathrm{in}}}{F}\right)\mathbf{U}_k + \sum_{\substack{j=1 \\ m_{ip_j}<0}}^{n_{ip(\mathrm{SCV})}} \alpha_j \mathbf{u}(ip_j), \tag{8}$$

where $m_{\mathrm{in}} = \sum_{\substack{j=1 \\ m_{ip_j}<0}}^{n_{ip(\mathrm{SCV})}} - m_{ip_j}$ denotes the amount of mass flowing into the subcontrol volume, F is the maximum between the amount of mass flowing in (m_{in}) and the mass flowing out $\left(m_{\mathrm{out}} = \sum_{\substack{j=1 \\ m_{ip_j}<0}}^{n_{ip(\mathrm{SCV})}} m_{ip_j}\right)$ and $\alpha_j = -\frac{\mathbf{u}(ip_j)\mathbf{n}(ip_j)}{F}$ are the weighting factors. For the example in Fig. 3 only the mass flow at integration point ip_3 is important to determine the upwind point.

3.1.2. Diffusion

The diffusion operator mainly determines the stabilisation constant introduced in the continuity equation. Because of that fact it will be thoroughly discussed. In Fig. 4 two possibilities to derive a Laplacian are indicated. For the case illustrated in Fig. 4a this leads to:

$$\begin{aligned}\Delta\phi(ip) &\approx \frac{\phi^{x^+} - 2\phi + \phi^{x^-}}{\frac{1}{4}\Delta x^2} + \frac{\phi^{y^+} - 2\phi + \phi^{y^-}}{\frac{1}{16}\Delta y^2} = \frac{\frac{3}{4}\Phi_1 + \frac{1}{4}\Phi_2 - 2\phi + \frac{3}{4}\Phi_0 + \frac{1}{4}\Phi_3}{\frac{1}{4}\Delta x^2} + \frac{\frac{1}{4}\sum_{k=1}^{n_N}\Phi_k - 2\phi + \frac{1}{2}(\Phi_0 + \Phi_1)}{\frac{1}{16}\Delta y^2} \\ &= \underbrace{\left(\frac{8}{\Delta x^2} + \frac{32}{\Delta y^2}\right)}_{:=(L^2_{d_{(1)}})^{-1}} \left(\underbrace{\frac{3}{8}(\Phi_0 + \Phi_1) + \frac{1}{8}(\Phi_2 + \Phi_3)}_{=\sum_{k=1}^{n_N} N_k(ip)\Phi_k} - \phi \right) = \frac{1}{L^2_{d_{(1)}}} \left(\sum_{k=1}^{n_N} N_k(ip)\Phi_k - \phi \right).\end{aligned} \tag{9}$$

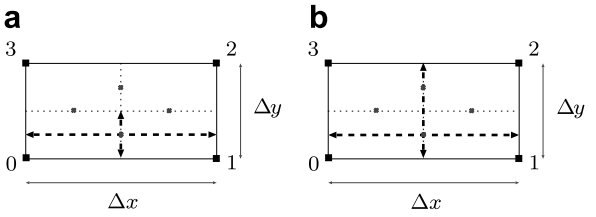

Fig. 4. Two possibilities to approximate the Laplacian at an integration point: (a) ansatz (1); (b) ansatz (2).

$x^{\pm}$, $y^{\pm}$ correspond to the positions indicated by the arrows in Fig. 4. Again, capital letters denote nodal values, whereas small letters are integration point values or interpolated values at the indicated positions. Correspondingly, the result for the second method in Fig. 4b is

$$\begin{aligned}\Delta\phi(ip) &\approx \frac{\phi^{x^+}-2\phi+\phi^{x^-}}{\frac{1}{4}\Delta x^2}+\frac{\frac{\phi^{y^+}-\phi}{\|y^+-ip\|}-\frac{\phi-\phi^{y^-}}{\|ip-y^-\|}}{\frac{1}{2}\|y^+-y^-\|}=\frac{\frac{3}{4}\Phi_1+\frac{1}{4}\Phi_2-2\phi+\frac{3}{4}\Phi_0+\frac{1}{4}\Phi_3}{\frac{1}{4}\Delta x^2}+\frac{\frac{1}{2}(\Phi_2+\Phi_3)-\phi}{\frac{3}{8}\Delta y^2}-\frac{\phi-\frac{1}{2}(\Phi_0+\Phi_1)}{\frac{1}{8}\Delta y^2}\\ &=\left(\tfrac{8}{\Delta x^2}+\tfrac{32}{3\Delta y^2}\right)\left(\tfrac{3}{8}(\Phi_0+\Phi_1)+\tfrac{1}{8}(\Phi_2+\Phi_3)-\phi\right)=\frac{1}{L^2_{d_{(2)}}}\left(\sum_{k=1}^{n_N}N_k(ip)\Phi_k-\phi\right).\end{aligned} \tag{10}$$

In three dimensions similar expressions can be derived, see [10]. As a first result the only difference between these two possibilities is the diffusion length scale L_d^2. In contrast to the above mentioned versions, Schneider and Raw [17] used the following length scale:

$$L^2_{d_{\mathrm{Raw}}}=\frac{1}{8}\Delta x^2+\frac{3}{32}\Delta y^2. \tag{11}$$

The behaviour of these three possibilities is different and, especially as the aspect ratio $\sigma=\frac{\Delta x}{\Delta y}$ tends to infinity, problems arise (Δx fixed)

$$\begin{aligned}\frac{1}{L^2_{d_{\mathrm{Raw}}}}&=\frac{1}{\frac{1}{8}\Delta x^2+\frac{3}{32}\Delta y^2}=\frac{32}{4\Delta x^2+3\Delta y^2}\overset{\sigma\to\infty}{\to}\frac{8}{\Delta x^2},\\ \frac{1}{L^2_{d_{(2)}}}&=\frac{8}{\Delta x^2}+\frac{32}{3\Delta y^2}=\frac{32\sigma^2+24}{3\Delta x^2}\overset{\sigma\to\infty}{\to}\infty,\\ \frac{1}{L^2_{d_{(1)}}}&=\frac{8}{\Delta x^2}+\frac{32}{\Delta y^2}=\frac{32\sigma^2+8}{\Delta x^2}\overset{\sigma\to\infty}{\to}\infty.\end{aligned}$$

Thus, the above approaches from Eqs. (9) and (10) are unstable. Regarding the stationary Stokes equation without convection, Eq. (6) in combination with ansatz (2) (and ansatz (1) respectively) tends to:

$$-v\frac{\sum_{k=1}^{n_N}N_k(ip)U_i^k-u_i}{L_d^2}+\sum_{k=1}^{n_N}\frac{\partial N_k}{\partial x_i}P_k=0\iff u_i=\sum_{k=1}^{n_N}N_k(ip)U_i^k-\frac{L_d^2}{v}\sum_{k=1}^{n_N}\frac{\partial N_k}{\partial x_i}P_k\overset{L_d^2\to 0}{\to}u_i=\sum_{k=1}^{n_N}N_k(ip)U_i^k.$$

This means, stabilisation is lost and the resulting system can not be solved. Analogous results can be derived for the three-dimensional case too, see [10]. To prevent this behaviour, a modification of ansatz (2) is introduced, which can be formulated in the following way. First, for generalisation to the unstructured case, the diffusion length scales at an *ip* can be expressed by the normal vector $\mathbf{n}$ belonging to that integration point (scaled with the corresponding area of the face) and the subcontrol volume measure |SCV|

$$L^2_{d_{\mathrm{Raw}}}=\frac{|\mathrm{SCV}|^2}{2\|\mathbf{n}\|^2}+\frac{3}{8}\|\mathbf{n}\|^2,\quad \frac{1}{L^2_{d_{(2)}}}=\frac{2\|\mathbf{n}\|^2}{|\mathrm{SCV}|^2}+\frac{8}{3\|\mathbf{n}\|^2}.$$

Then the expression for ansatz (2) can be stabilised by

$$\frac{1}{L^2_{d_{(2)_{\mathrm{corr}}}}}=\frac{2\|\mathbf{n}_{\min}\|^2}{|\mathrm{SCV}|^2}+\frac{8}{3\|\mathbf{n}_{\mathrm{avg}}\|^2} \tag{12}$$

and will be further referred to by method $(2)_{\mathrm{corr}}$. It depends on $\|\mathbf{n}_{\min}\|^2=\min_{\mathbf{n}\in\mathrm{element}}\|\mathbf{n}\|^2$ the minimal norm of all normals of that element and on $\|\mathbf{n}_{\mathrm{avg}}\|^2=\frac{1}{n_N}\sum_{k=1}^{n_N}\|\mathbf{n}_k\|^2$ an average measure of the normals. A bounded expression results which can be proved for the orthogonal grid case

$$\frac{1}{L^2_{d_{(2)_{\mathrm{corr}}}}}=\frac{8}{\Delta x^2}+\frac{64}{3(\Delta x^2+\Delta y^2)}=\frac{88+24\frac{1}{\sigma^2}}{3\Delta x^2(1+\frac{1}{\sigma^2})}\overset{\sigma\to\infty}{\to}\frac{88}{3\Delta x^2}.$$

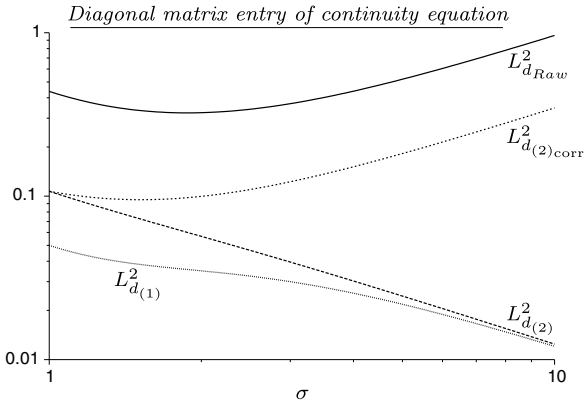

Fig. 5. Evolution of the diagonal matrix entry for different diffusion length scale approximations.

This is illustrated in Fig. 5 by the behaviour of the diagonal matrix entry of the continuity equation over the aspect ratio. But first of all, the result of the stabilisation FIELDS will be presented.

3.1.3. Result for the stationary Stokes equation

At each integration point an interpolation for the velocity u_i can be derived which has to be inserted in Eq. (2). Summation over all subcontrol volumes belonging to one node the stencil of the continuity equation for the pressure dependence in the interior looks like

$$\begin{bmatrix} -\frac{1}{8}\left(L^2_{d_1}\frac{1}{\sigma}+L^2_{d_2}\sigma\right) & -\frac{1}{4}\left(-L^2_{d_1}\frac{1}{\sigma}+L^2_{d_2}3\sigma\right) & -\frac{1}{8}\left(L^2_{d_1}\frac{1}{\sigma}+L^2_{d_2}\sigma\right) \\ -\frac{1}{4}\left(L^2_{d_1}\frac{3}{\sigma}-L^2_{d_2}\sigma\right) & \frac{3}{2}\left(L^2_{d_1}\frac{1}{\sigma}+L^2_{d_2}\sigma\right) & -\frac{1}{4}\left(L^2_{d_1}\frac{3}{\sigma}-L^2_{d_2}\sigma\right) \\ -\frac{1}{8}\left(L^2_{d_1}\frac{1}{\sigma}+L^2_{d_2}\sigma\right) & -\frac{1}{4}\left(-L^2_{d_1}\frac{1}{\sigma}+L^2_{d_2}3\sigma\right) & -\frac{1}{8}\left(L^2_{d_1}\frac{1}{\sigma}+L^2_{d_2}\sigma\right) \end{bmatrix}_P . \tag{13}$$

$L^2_{d_1}$ is the diffusion length scale belonging to the vertical subcontrol volume surfaces and $L^2_{d_2}$ represents the length scale for the horizontal subsurfaces. If the length scale from Eq. (12) is taken then $L^2_{d_1} = L^2_{d_2}$, but for Eq. (11) they are different. This means, for the first one the stabilisation is of the form of a Laplacian $-L^2_d\Delta p$ while for the second one an anisotropic Laplacian is introduced $-\left(L^2_{d_1}\frac{\partial^2 p}{\partial x^2}+L^2_{d_2}\frac{\partial^2 p}{\partial y^2}\right)$. Therefore the behaviour of the different stabilisations resulting from different diffusion length scale approximations differ not only by a constant. To illustrate this fact, the diagonal matrix entry of the continuity equation is plotted in Fig. 5. The behaviour of Raw's method in comparison to version (12) is similar, but differs slightly. Also the rather fast loss of the stabilisation for the diffusion length scales $L^2_{d_{(1)}}$ and $L^2_{d_{(2)}}$ can be clearly seen. Already an aspect ratio around 4 is a hard problem to solve. This can be demonstrated by a very simple test problem, the Poiseuille flow for the Stokes equations which will be presented in Section 5.1. Fig. 5 also shows the different constants involved by the various methods. It looks like as if ansatz (1) produces the smallest amount of error in the continuity equation. But a numerical example will show, this is not the case because the stabilising effect is too low for this choice. The version of Raw produces a rather big constant in comparison to ansatz (2), which can also be seen in numerical experiments. The method $(2)_{\mathrm{corr}}$ shows about the same behaviour as Raw's method as the aspect ratio increases, although there is a fundamental difference between these two, not only in the resulting constant but also in the fact, that one leads to a Laplacian while the other introduces an anisotropic Laplace-operator as disturbance in the continuity equation. Thus, these two methods differ in their overall behaviour. This fact we will address again later.

3.1.4. Result for the stationary Navier–Stokes equation

To determine the stencil for the nonlinear case, we assume a uniform velocity field $\tilde{\mathbf{u}} = (1,0)^{\mathrm{T}}$ and choose the LPS method as upwind scheme. Then we get the following approximation at each integration point:

$$u_i = \left(\frac{\nu}{L_d^2} + \frac{1}{L_c}\right)^{-1} \left(\frac{\nu}{L_d^2} \sum_{k=1}^{n_N} N_k U_i^k + \frac{1}{L_c} u_i^{\mathrm{up}} - \sum_{k=1}^{n_N} \frac{\partial N_k}{\partial x_i} P^k\right). \tag{14}$$

The following stencil for the pressure dependence arises:

$$\begin{bmatrix} -\frac{1}{8}\left(F_1 \frac{1}{\sigma} + F_2\sigma\right) & -\frac{1}{4}\left(-F_1 \frac{1}{\sigma} + \frac{3}{2}(F_2+F_3)\sigma\right) & -\frac{1}{8}\left(F_1 \frac{1}{\sigma} + F_3\sigma\right) \\ -\frac{1}{4}\left(F_1 \frac{3}{\sigma} - F_2\sigma\right) & \frac{3}{2}\left(F_1 \frac{1}{\sigma} + \frac{1}{2}(F_2+F_3)\sigma\right) & -\frac{1}{4}\left(F_1 \frac{3}{\sigma} - F_3\sigma\right) \\ -\frac{1}{8}\left(F_1 \frac{1}{\sigma} + F_2\sigma\right) & -\frac{1}{4}\left(-F_1 \frac{1}{\sigma} + \frac{3}{2}(F_2+F_3)\sigma\right) & -\frac{1}{8}\left(F_1 \frac{1}{\sigma} + F_3\sigma\right) \end{bmatrix}_P,$$

which is slightly different from the linear case above due to the convection term, but mainly the same structure is retained. The F_i are of the form $F_i = \left(\frac{\nu}{L_{d_j}^2} + \frac{1}{L_{c_i}}\right)^{-1}$ depending on the diffusion and convection length scales: $L_{c_1} = \frac{1}{2}\Delta x$, $L_{c_2} = \frac{3}{4}\Delta x$, $L_{c_3} = \frac{1}{4}\Delta x$ and $L_{d_j}^2$ as above for the Stokes case.

3.2. FLOW

Now the differences of the FLOW scheme in comparison to the FIELDS method will be addressed. Karimian demonstrated that for the low compressible regime the FIELDS method still allows oscillating solutions, see e.g. [15]. To prevent this behaviour he introduced a modification which also takes the continuity equation into account to determine the special interpolation for the stabilisation. In the FIELDS method, only the momentum equation is considered disregarding the error in mass conservation to create the stabilisation. Karimians modification demands that the momentum equation error $\dot{\epsilon}$ and the error of the continuity equation $\ddot{\epsilon}$ are balanced in all integration points in the following way:

$$\underbrace{\frac{\partial u_i}{\partial t} - \nu\Delta u_i + \|\tilde{\mathbf{u}}\|\frac{\partial u_i}{\partial s} + \frac{\partial p}{\partial x_i}}_{:=\dot{\epsilon}} = u_i \underbrace{\mathrm{div}(\mathbf{u})}_{:=\ddot{\epsilon}}\Bigg|_{ip_l}.$$

The mass error $\ddot{\epsilon}$ times u_i can be reformulated to

$$u_i\ddot{\epsilon} = \sum_{j=1}^{d} u_j \frac{\partial u_i}{\partial x_j} - \sum_{j=1, j\neq i}^{d} \left(u_j \frac{\partial u_i}{\partial x_j} - u_i \frac{\partial u_j}{\partial x_j}\right).$$

This can be approximated and discretised by

$$u_i\ddot{\epsilon} \approx \|\tilde{\mathbf{u}}\| \frac{u_i^{\mathrm{dn}} - u_i^{\mathrm{up}}}{L_{\mathrm{dn}}} - \sum_{j=1, j\neq i}^{d} \sum_{k=1}^{n_N} \left(\tilde{u}_j \frac{\partial N_k}{\partial x_j} U_i^k - \tilde{u}_i \frac{\partial N_k}{\partial x_j} U_j^k\right),$$

where L_{dn} denotes the length between the upwind and downwind point. This leads to the FLOW approximation for the integration point velocities:

$$\frac{u_i - u_i^0}{\Delta t} - \nu \frac{\sum_k N_k U_i^k - u_i}{L_d^2} + \|\tilde{\mathbf{u}}\| \frac{u_i - u_i^{\mathrm{up}}}{L_c} + \sum_{k=1}^{n_N} \frac{\partial N_k}{\partial x_i} P^k - \|\tilde{\mathbf{u}}\| \frac{u_i^{\mathrm{dn}} - u_i^{\mathrm{up}}}{L_{\mathrm{dn}}} + \sum_{j=1, j\neq i}^{d} \sum_{k=1}^{n_N} \left(\tilde{u}_j \frac{\partial N_k}{\partial x_j} U_i^k - \tilde{u}_i \frac{\partial N_k}{\partial x_j} U_j^k\right) = 0. \tag{15}$$

In contrast to the approximation of FIELDS in Eq. (6), the resulting integration point velocities of the FLOW-ansatz depend on all velocity components in the nodes which leads to a tighter coupling of the components and to a better nonlinear behaviour, see examples in Sections 5.2 and 5.3. Especially if strong vortices are present in the flow the nonlinear convergence of the algorithm in combination with the FLOW stabilisation is much better in comparison to the FIELDS stabilisation. But also the error in the continuity equation tends to be lower than for FIELDS which will be shown in numerical examples. Thus, also for the incompressible regime the FLOW stabilisation improves the results.

3.3. Finite volume approximation of convection

A very important contribution to the overall behaviour of the discretisation comes from the discretisation of the nonlinear term in the momentum equation (compare Eq. (3))

$$\sum_{j=1}^{d} u_i \tilde{u}_j n_j.$$

Several possibilities to approximate the unknown integration point velocity u_i can be used.

Central differences: These are realized by

$$u_i = \sum_{k=1}^{n_N} N_k U_i^k \tag{16}$$

but are not suited for convection dominated problems, since they introduce a downwind dependency which leads to unphysical solutions.

Upwinding: This can be done by using one of the upwind techniques already described in Section 3.1.1

$$u_i \approx u_i^{\mathrm{up}}, \tag{17}$$

where the upwind velocity has to be interpolated at the upwind position. The method is suited for strong convection but has an immense dependence on the artificial diffusion introduced by the various upwind methods.

Physical advection correction (PAC): PAC-upwinding can be written in the form

$$u_i(ip) \approx \breve{u}_i^{\mathrm{up}} + \delta u_i \tag{18}$$

which is a generalised skewed upwinding technique introduced by Raithby [12] depending on a scaled upwind velocity plus a correction term. In the context of this paper the special interpolation already described above in Eq. (6) can be used leading to the following approximations for the upwind point and the correction term:

$$\breve{u}_i^{\mathrm{up}} = \left(\frac{1}{\Delta t} + \frac{\nu}{L_d^2} + \frac{\|\tilde{\mathbf{u}}\|}{L_c}\right)^{-1} \frac{\|\tilde{\mathbf{u}}\|}{L_c} u_i^{\mathrm{up}},$$

$$\delta u_i = \left(\frac{1}{\Delta t} + \frac{\nu}{L_d^2} + \frac{\|\tilde{\mathbf{u}}\|}{L_c}\right)^{-1} \left(\frac{u_i^0}{\Delta t} + \frac{\nu}{L_d^2} \sum_{k=1}^{n_{co}} N_k U_i^k - \sum_{k=1}^{n_{co}} \frac{\partial N_k}{\partial x_i} P^k\right).$$

This method shows a very good behaviour and increases the order of the upwind technique inside the interpolation clearly as will be shown by means of the Driven Cavity problem.

Additionally, depending on the local Peclet number Pe ($Pe = \frac{u_n L}{\nu}$, normal velocity u_n, local length scale L) a blending between central differences and one of the two upwind techniques described above is always used to increase the approximation order of the convection operator.

4. Solver settings and realisation

The solving strategy will be described very briefly. A quasi-Newton iteration is used as nonlinear solver, leading to a fixed point iteration. In each nonlinear step the resulting fully coupled linear equation system is solved by a standard geometric multigrid method. The full weighting operator serves as restriction, while for prolongation bilinear interpolation is used. The multigrid employs point-block ILU_β as smoother, based on $(d+1) \times (d+1)$-blocks at each grid point whereas the ordering of the unknowns in these blocks is $u_1,\ u_2(,u_3),\ p$. A V-cycle with different settings for β, pre- and post-smoothing steps is used. The solver settings worked well for all numerical examples, especially since the multigrid method is very robust and well suited for these kind of applications. But the focus of this paper is on the discretisation part, therefore the behaviour of the solver for the various test cases will not be documented.

The methods described in Section 3 have been implemented in the UG framework. UG is a software system for the simulation of PDE-based models providing a lot of advanced numerical features. The main simulation

strategies are the combination of adaptivity on locally refined unstructured grids, parallelism aiming at the use of massively parallel computers and multigrid methods, [3,4,5]. Thus, it is very easy to run the applications in parallel, since the assembly of the stiffness matrix can be done element-wise and since each element is uniquely assigned to one processor. Therefore, all the features dealing with parallelisation can be left to the core part of UG but (almost) no adjustments have to be done on the application side.

5. Numerical results

In this section, different aspects of the discretisation already mentioned in the previous sections together with criteria for comparison and judgement of the presented methods will be demonstrated by various numerical experiments.

For the first example an analytical solution is known, therefore it is very easy to determine the error introduced by the discretisation. The influence of the diffusion length scales will be thoroughly studied for this problem. But due to the rather simple flow structure no severe differences arising from the two stabilisations FIELDS and FLOW are expected. To study their influence more complex flow problems are investigated like the Driven Cavity flow and the flow around a cylinder. For the second example, the Driven Cavity problem, no analytical solution is known but reference data is available. Thus, the deviation from the data can be shown to demonstrate the influence of upwind techniques especially. Finally, the last example will compare all stabilisation methods in combination with upwind techniques and diffusion length scales.

5.1. Poiseuille flow

A laminar incompressible flow between two parallel walls at $y = \pm a$ is called Poiseuille flow. The analytic solution for this problem in 2d is given by

$$u_1(x,y) = \frac{1}{2}\frac{c}{v}(y^2 - a^2), \quad u_2(x,y) = 0, \quad p(x,y) = cx + p_0,$$

where p_0 is fixed by the zero outflow boundary condition for pressure (see e.g. [18]). The parabolic profile of the analytic solution is prescribed at the inflow and no-slip boundary conditions are set at the walls. For simplicity, the parameters are chosen as follows: $a = 1$ and $u_{\max} = -\frac{1}{2}\frac{c}{v}a^2 = 1.5$ leading to a mean velocity of 1 at the inlet. The length of the channel is set to 10 from what follows $p_0 = -cx$ where c is fixed by the choice for v.

This example is well suited to study the effect of different diffusion length scales presented in Section 3.1.2. Since an analytical solution is known the error of the solution is computed to analyse the methods. Therefore the first criterion is the error of the total solution defined by

$$E = \sqrt{\sum_{j=1}^{d+1} \|c_j - c_j^{\text{exact}}\|_2} \quad \text{where } c_j = u_1,\ u_2 \text{ or } p.$$

But also the error of mass conservation is always very important for fluid dynamics, thus

$$M = \|\text{div}(\mathbf{u})\|_2$$

serves as second criterion. First, the linear case at $Re = 1$ $\left(Re = \frac{1 \cdot a}{v}\right)$ is investigated, afterwards also the nonlinear case is addressed.

5.1.1. The stokes case, Re = 1

Since we examine the linear case, the nonlinear coupling introduced by the FLOW stabilisation would not be appropriate and would not be regarded for this case. But, the behaviour of the discretisation if the various diffusion length scales presented in Section 3.1.2 are used can be demonstrated, especially as the grid aspect ratio is increased.

First the results for a cartesian grid with mesh size $h = \Delta x = \Delta y = \frac{1}{2}$ are presented (Table 1). Afterwards the aspect ratio is increased by enlarging the mesh width along the x-axis from $\sigma = 1$ to $\sigma = 4$. For all test cases,

Table 1
Poiseuille: mass and solution error for cartesian grid

Level		1	2	3	4	5
M	Raw	0.00240432	0.000219896	1.97396e−05	1.75814e−06	1.56213e−07
	(1)	0.00151363	0.000136021	1.21216e−05	1.07575e−06	9.52405e−08
	(2)	0.00085879	7.74391e−05	6.91208e−06	6.13905e−07	5.43790e−08
E	Raw	1.80553	0.461718	0.116553	0.0293914	0.00744383
	(1)	1.06952	0.270761	0.070228	0.0188056	0.00537335
	(2)	0.60655	0.152359	0.038666	0.0099527	0.00263496

the coarsest grid is uniformly refined five times and the evolution of the error E and the error of mass conservation M is presented on all levels.

Although the entry on the diagonal in the continuity equation is smallest for ansatz (1) as diffusion length scale (see Fig. 5) the resulting error of mass conservation is higher than for ansatz (2). But in accordance with the plot of the diagonal entry in the continuity equation, the mass conservation is better for ansatz (2) than Raw's method. The rather fast loss of the stabilisation if the length scales resulting from ansatz (1) and (2) are used can be seen in Tables 2 and 3. There the method $(2)_{\text{corr}}$ from (12) is also considered. Clearly, the error for this stabilised method is higher than for method (2) but still much smaller than the method of Raw. Already for a very low aspect ratio of 2 the solver diverged on level 5 for version (1) and is therefore omitted for the $\sigma = 4$ case. Similarly, the behaviour of method (2) degenerates for the next aspect ratio $\sigma = 4$ as well due to the loss of stabilisation indicated by Fig. 5. The solver diverges even on level 3 because the resulting equation system does not include an appropriate coupling between velocity and pressure and leads to an unsolvable system. But method $(2)_{\text{corr}}$ still works well and produces a much lower error in mass and solution in comparison to the method of Raw.

This investigation demonstrates the rather big influence of diffusion length scales on the results, especially on the resulting mass error and the amount of stabilisation introduced in the equation system.

Table 2
Poiseuille: mass and solution error for aspect ratio $\sigma = 2$

Level		1	2	3	4	5
M	Raw	0.0132922	0.00125770	0.000113965	1.01891e−05	9.05667e−07
	(1)	0.0020905	0.00018555	1.64404e−05	1.45587e−06	–
	(2)	0.0004643	4.16656e−05	3.71114e−06	3.29289e−07	2.91612e−08
	$(2)_{\text{corr}}$	0.0019916	0.00018052	1.61463e−05	1.43555e−06	1.27256e−07
E	Raw	6.74779	1.811270	0.462930	0.1169820	0.0295798
	(1)	1.04904	0.274445	0.074976	0.0216881	–
	(2)	0.22507	0.057108	0.014689	0.0038696	0.0010586
	$(2)_{\text{corr}}$	0.97029	0.246908	0.062668	0.0160388	0.0041804

Table 3
Poiseuille: mass and solution error for aspect ratio $\sigma = 4$

Level		1	2	3	4	5
M	Raw	0.0625307	0.00680875	0.000638435	5.76395e−05	5.14441e−06
	(2)	0.0024513	0.00014461	–	–	–
	$(2)_{\text{corr}}$	0.0195004	0.00183176	0.000164923	1.4699e−05	1.30458e−06
E	Raw	21.4702	6.76111	1.81313	0.4634	0.117165
	(2)	0.22728	0.052959	–	–	–
	$(2)_{\text{corr}}$	6.50919	1.7418	0.446364	0.113669	0.0292003

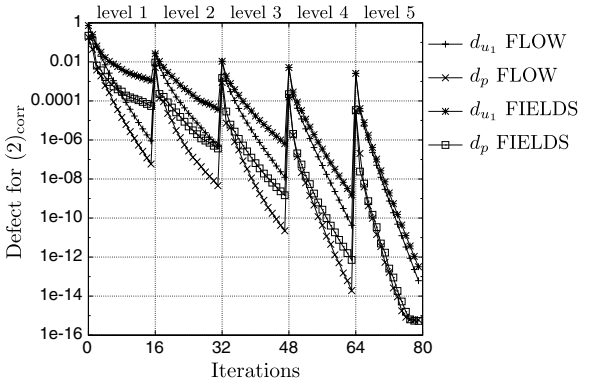

Fig. 6. Defect reduction for the Poiseuille problem at $Re = 100$ for the u_1 and p component.

5.1.2. The nonlinear case at Re = 100

For the nonlinear case at $Re = 100$ a grid stretched in y-direction with a maximum aspect ratio of $\sigma = 8.6$ is used and both stabilisations, FIELDS as well as FLOW, are investigated. But comparing the two stabilisation methods does not expose severe discrepancies. Due to the character of the flow no strong coupling between the velocity components is necessary, therefore almost no differences can be recognised besides the nonlinear behaviour and defect reduction. There the tighter coupling of FLOW comes into play which leads to a better nonlinear convergence as can be seen in Fig. 6, where the defects on each grid level inside a nested iteration are plotted versus the iterations.

The nonlinear convergence behaviour serves as third criterion as the nonlinear defect reduction is crucial for the nonlinear process.

5.2. Driven cavity

This test case is dedicated to the investigation of the convection term treatment in the momentum equation. In this term a lot of numerical diffusion can be introduced which destroys the approximation quality severely. By comparison with reference data the amount of numerical diffusion of the methods described in Section 3.3 can be visualised. The Driven Cavity problem is furthermore complicated enough to demonstrate the difficulties of the stabilisation methods.

The problem is defined in the unit square with no-slip boundary conditions everywhere except the top lid. There the velocity is prescribed to be $\mathbf{u} = (1, 0)^{\mathrm{T}}$. A relative high Reynolds number of $Re = 3200$ based on the lid velocity and the side length of the domain is chosen. Two upwind schemes (LPS and POS, see Section 3.1.1) in combination with physical advection correction or standard upwinding from Section 3.3 are investigated. The results are compared to reference values cited in [7], the values of u_1 along $x = \frac{1}{2}$ and u_2 along $y = \frac{1}{2}$. The coarsest grid and the resulting streamlines can be seen in Fig. 7. In contrast to the Poiseuille flow above, the

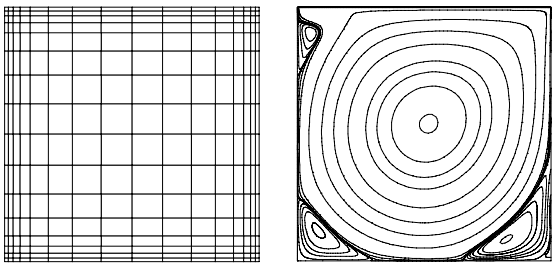

Fig. 7. Coarse grid and streamlines of the Driven Cavity problem.

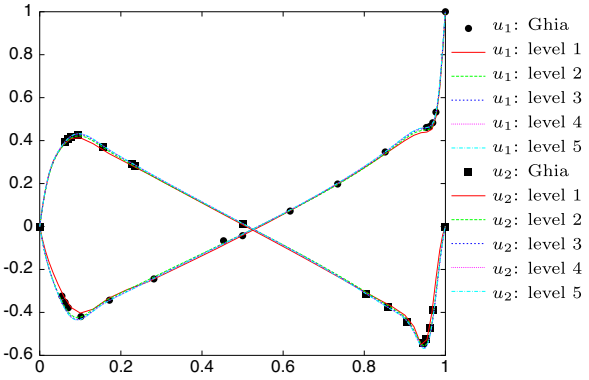

Fig. 8. Comparison with reference data for PAC-upwinding with LPS.

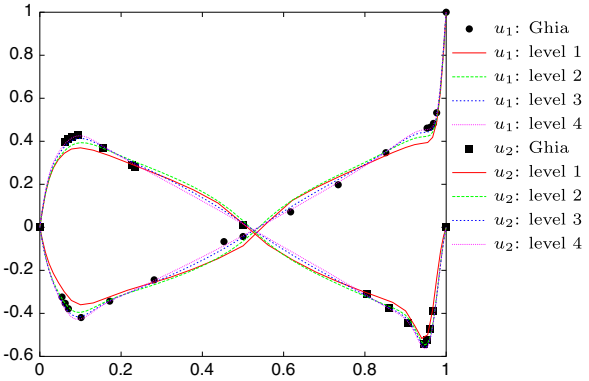

Fig. 9. Comparison with reference data for PAC-upwinding with POS.

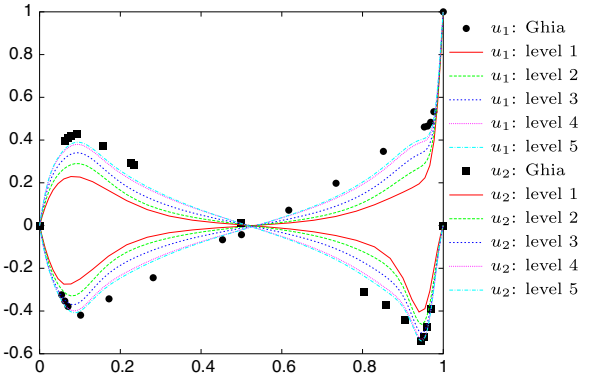

Fig. 10. Comparison with reference data for upwinding with LPS.

effect of different stabilisation types and convection treatment has a very strong influence on the resulting solution.

Again, the coarse grid is uniformly refined 5 times and the results on each level are illustrated in Figs. 8–10. First, the PAC-scheme from Eq. (18) in combination with the LPS-scheme (see Eq. (7)) leads to a very good approximation already on level 3. Further refinement is not necessary for this scheme, since the results match the reference data very well. If standard upwinding from Eq. (17) in combination with LPS is used, the results

Table 4
Evolution of error in mass conservation for the Driven Cavity problem

Level		1	2	3	4
Raw	POS	5.73515e−05	8.18035e−06	7.0311e−07	9.62171e−08
	LPS	4.31159e−05	3.91027e−06	5.02157e−07	8.48433e−08
$(2)_{corr}$	POS	5.43548e−05	5.10849e−06	6.57831e−07	1.33205e−07
	LPS	3.53489e−05	3.12022e−06	5.58418e−07	1.27753e−07

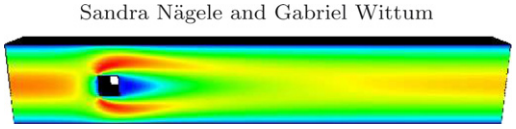

Fig. 11. Solution of flow around cylinder: u_1-component.

are much worse and a much higher resolution is necessary to get approximately the same accuracy. This demonstrates that advection correction increases the approximation order explicitly, whereas a simple upwind technique leads to a rather high numerical diffusivity as Fig. 10 demonstrates. But the result can get worse if physical advection correction is combined with a very diffusive scheme like the POS-scheme from Eq. (8). For some situations POS is identical to the full upwind scheme which is known to be very robust, but very diffusive too. In combination with POS, the PAC-scheme also delivers a rather bad solution illustrated in Fig. 9. One can draw the conclusion that it is very important which upwind technique is used for the physical advection correction scheme, otherwise the resulting solution would not be any better than a solution with a standard upwind scheme like the one shown in Fig. 10. Especially if vortices are present, the differences resulting from upwind schemes become eminent.

At last, the resulting error in mass conservation is listed in Table 4 for the PAC-upwind method in combination with LPS, POS and the diffusion length scales from Raw and $(2)_{corr}$. For this case, the method $(2)_{corr}$ results in a slightly worse behaviour of mass conservation which can be explained qualitatively as follows. The stabilisation results in an expression like

$$\operatorname{div}(\mathbf{u}) - Ch^2\Delta p = 0 \iff s\Delta p = \frac{1}{Ch^2}\operatorname{div}(\mathbf{u})$$

as continuity equation. For method $(2)_{corr}$ the constant C is lower, but the curvature is higher and therefore the pressure gradients in the upper two corners get stronger and are less well resolved by the bilinear ansatz functions. As a consequence, the coupling between velocity and pressure is reduced which leads to a higher error in mass conservation.

All of the above combinations for convection treatment and diffusion length scale were run for the FLOW stabilisation. The FIELDS method did not lead to an equation system which could be solved properly, at least not if PAC-upwinding was used. But since the standard upwind method does not result in a good approximation of the solution, the data for the FIELDS method were not shown. FLOW is more robust in that sense, which justifies its application for the incompressible case, although it was designed for the weakly compressible regime primarily.

5.3. Flow around a cylinder

The last test case examines the differences and effects of the two stabilisation techniques presented, FIELDS and FLOW, in combination with upwind methods and diffusion length scales.

The problem is a benchmark problem published in [14], a three-dimensional test case of a flow around a cylinder with square cross-section. As inflow a parabolic profile is prescribed, no-slip boundary conditions are set at the walls and a zero pressure boundary condition is prescribed at the outflow. In Fig. 11, the resulting u_1-component of the velocity is plotted. The colour varies between blue (lowest value) and red (highest

Table 5
Flow around cylinder: error of mass conservation on different grid levels

Level			1	2	3	4
FIELDS	Raw	POS	2.08591e−05	1.97202e−06	1.77348e−07	1.71119e−08
		LPS	1.81634e−05	1.61009e−06	1.50132e−07	1.88796e−08
	$(2)_{corr}$	POS	1.81415e−05	1.32937e−06	9.86493e−08	9.40105e−09
		LPS	1.44863e−05	1.08914e−06	8.81611e−08	9.47087e−09
FLOW	Raw	POS	1.48164e−05	1.6113e−06	1.52455e−07	1.43695e−08
		LPS	1.51535e−05	1.44984e−06	1.37332e−07	1.69309e−08
	$(2)_{corr}$	POS	1.30678e−05	1.06724e−06	8.42097e−08	7.25313e−09
		LPS	1.28927e−05	1.02257e−06	8.4783e−08	8.89071e−09

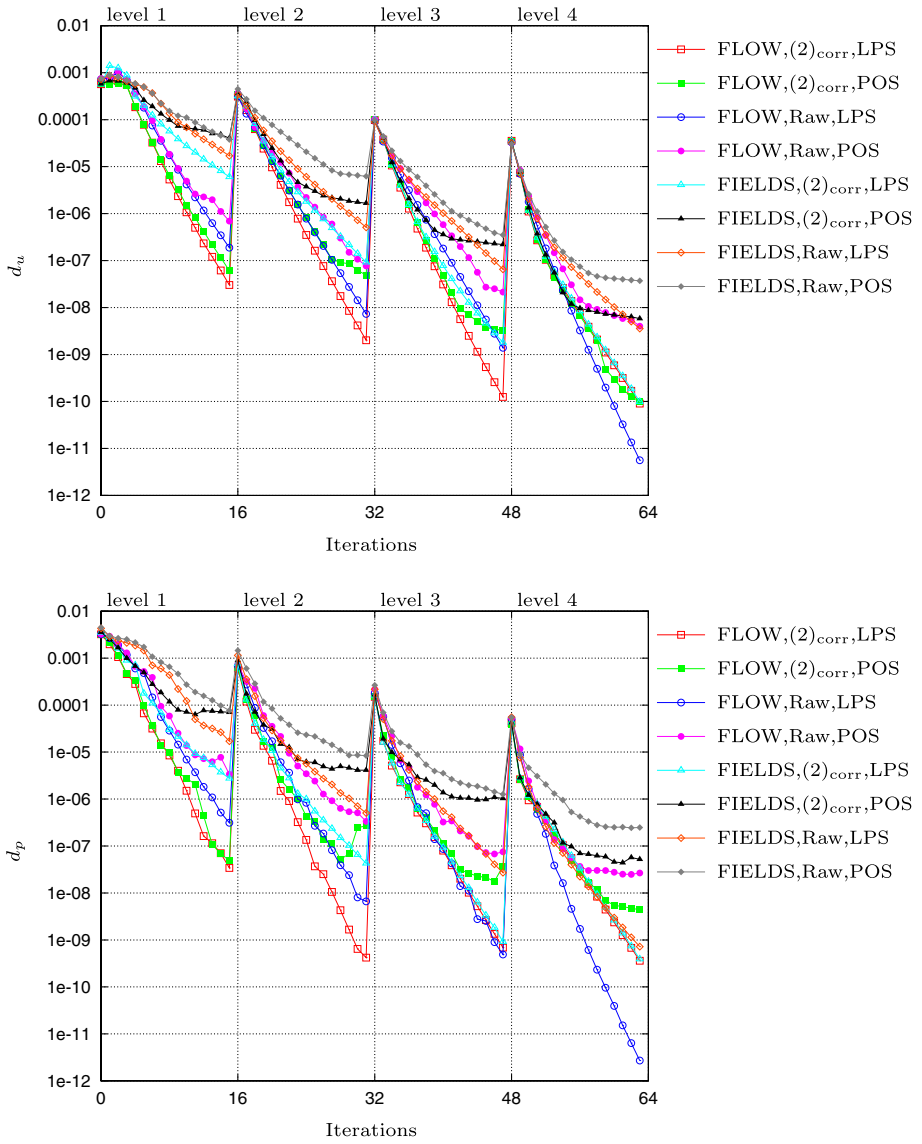

Fig. 12. Flow around cylinder: nonlinear defect versus grid levels and iterations.

value).[1] It is a stationary solution where the different behaviour of the stabilisations can be investigated. The coarse grid consists of 168 hexahedra and is uniformly refined four times. Thus, the finest grid has nearly three million unknowns. Because of the problem size, this test case was run on 16 processors to speed up run time using the parallel features of the UG framework.

First, the mass conservation property is studied. For this, various runs were performed and the resulting error is listed in Table 5. As already mentioned in Section 5.1, the resulting mass conservation error is less for diffusion length scale $(2)_{\text{corr}}$ although it is not as clear as for the Stokes-problem, but still an obvious difference can be observed. But more important is the difference between FLOW and FIELDS, since FLOW leads to a slightly better approximation in comparison to the corresponding FIELDS methods, which is at least a small improvement and justifies again the application to the incompressible regime.

But a rather big and important improvement is the nonlinear convergence of the resulting equation system for the FLOW stabilisation which can be demonstrated by Fig. 12. There, the nonlinear defect is plotted for each grid level inside a nested iteration. The important feature is the better nonlinear convergence rate of the FLOW stabilisation in contrast to the corresponding FIELDS methods. This is a direct consequence of the tighter coupling between the velocity components and the pressure. FIELDS simply leads to a coupling of each velocity component to the pressure, but not to the other velocity components. FLOW does include such a inter-velocity dependence and therefore results in a better nonlinear behaviour. Furthermore, the defect reduction especially for the pressure component, is slowing down for the version including POS-upwinding. Thus, POS does not only lead to a poor solution compared to the LPS-method for instance (see discussion of the Driven Cavity problem), but also to a poorer convergence behaviour.

6. Conclusion

Two stabilisation techniques for a colocated finite volume method were presented. Both are based on similar ideas, but lead to different couplings between the components. The first one called FIELDS couples the velocities with the pressure and vice versa. Whereas the second one called FLOW also includes a coupling of the velocities to each other, which leads to a reduced error of mass conservation in comparison to FIELDS, but also to a much better nonlinear convergence. Especially for flows including vortices, like the Driven Cavity problem or turbulent cases which have not been discussed throughout this paper, the nonlinear behaviour is very important, otherwise essential flow features may be poorly resolved by the discretisation. In addition, for both stabilisation methods the strategy to approximate the diffusion length scale has a strong effect on the resulting error in mass conservation but also on the behaviour of the linear system and the linear solver. A very low constant introduced in the continuity equation does not necessarily lead to a low error, since the stabilising effect is also reduced which is essential for the colocated scheme. A kind of optimisation has to be done to find the lowest possible constant and through this the lowest perturbation of the mass equation but also a sufficiently high value to stabilise the equation system. This is a hard optimisation problem where it is not at all clear, if a unique solution exists which serves all desired purposes. But, at least a small improvement towards the optimal constant has been presented by the method $(2)_{\text{corr}}$. At last, the discretisation of the nonlinear term in the momentum equation is very important in view of the quality of the solution and the numerical diffusivity included there. But some combinations of the discretisation parameters, like diffusion length scale and upwind technique for the FIELDS method lead to an equation system which could not be solved. In this situations, the FLOW stabilisation proved to be very robust and fits better for problems involving vortices and/or singularities inherent in the flow. Thus, even for this incompressible application the method designed for weakly compressible flow problems includes important new features which were demonstrated experimentally.

References

[1] R. Alexander, Diagonally implicit Runge–Kutta methods for stiff O.D.E.'s, SIAM Journal on Numerical Analysis 14 (1977) 1006–1021.
[2] D.N. Arnold, F. Brezzi, M. Fortin, A stable finite element for the Stokes equations, Calcolo 21 (1984) 337–344.

[1] For interpretation of color in Fig. 11, the reader is referred to the web version of this article.

[3] P. Bastian, K. Birken, K. Johannsen, S. Lang, N. Neuss, H. Rentz-Reichert, C. Wieners, UG – a flexible software toolbox for solving partial differential equations, Computing and Visualization in Science 1 (1997) 27–40.
[4] P. Bastian, K. Birken, K. Johannsen, S. Lang, V. Reichenberger, C. Wieners, G. Wittum, C. Wrobel, A parallel software-platform for solving problems of partial differential equations using unstructured grids and adaptive multigrid methods, in: W. Jäger, E. Krause (Eds.), High Performance Computing in Science and Engineering '98, Springer, 1998, pp. 326–339.
[5] Peter Bastian, Klaus Johannsen, Stefan Lang, Sandra Nägele, Christian Wieners, Volker Reichenberger, Gabriel Wittum, Christian Wrobel, Advances in high-performance computing: multigrid methods for partial differential equations and its applications, in: High Performance Computing in Science and Engineering 2000, Springer, 2001, pp. 506–519.
[6] F. Brezzi, On the existence uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers, RAIRO Analyse Numerique 8 (1974) 129–151.
[7] U. Ghia, K.N. Ghia, C.T. Shin, High-*Re* solutions for incompressible flow using the Navier–Stokes equations and a multigrid method, Journal of Computational Physics 48 (1982) 387–411.
[8] Thomas J.R. Hughes, Leopold P. Franca, Marc Balestra, A new finite element formulation for computational fluid dynamics: V. circumventing the Babuška–Brezzi condition: a stable Petrov–Galerkin formulation of the Stokes problem accommodating equal-order interpolations, Computer Methods in Applied Mechanics and Engineering 59 (1986) 85–99.
[9] S.M.H. Karimian, G. Schneider, Pressure-based control-volume finite element method for flow at all speeds, AIAA Journal 33 (1995) 1611–1618.
[10] Sandra Nägele, Mehrgitterverfahren für die inkompressiblen Navier–Stokes Gleichungen im laminaren und turbulenten Regime unter Berücksichtigung verschiedener Stabilisierungsmethoden. Ph.D. thesis, Universität Heidelberg, 2003.
[11] Sandra Nägele, Gabriel Wittum, Large-eddy simulation and multigrid methods, Electronic Transactions on Numerical Analysis 15 (2003) 152–164.
[12] G.D. Raithby, Skew upstream differencing schemes for problems involving fluid flow, Computer Methods in Applied Mechanics and Engineering 9 (1976) 153–164.
[13] C.M. Rhie, W.L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, AIAA Journal 21 (1983) 1525–1532.
[14] M. Schäfer, S. Turek, Benchmark computations of laminar flow around a cylinder, in: E.H. Hirschel (Ed.), Flow Simulation with High-Performance Computers II, Notes on Numerical Fluid Mechanics, vol. 52, Vieweg, 1996, pp. 547–566.
[15] G.E. Schneider, S.M.H. Karimian, Advances in control-volume-based finite-element methods for compressible flows, Computational Mechanics 14 (1994) 431–446.
[16] G.E. Schneider, M.J. Raw, A. Skewed, Positive influence coefficient upwinding procedure for control-volume-based finite-element convection–diffusion computation, Numerical Heat Transfer 9 (1986) 1–26.
[17] G.E. Schneider, M.J. Raw, Control volume finite-element method for heat transfer and fluid flow using colocated variables – 1. Computational Procedure, Numerical Heat Transfer 11 (1987) 363–390.
[18] Pieter Wesseling, Principles of Computational Fluid Dynamics. Springer Series in Computational Mathematics, Springer, 2001.