INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 22, 987–1011 (1996)

A COMPARATIVE STUDY OF TIME-STEPPING TECHNIQUES FOR THE INCOMPRESSIBLE NAVIER–STOKES EQUATIONS: FROM FULLY IMPLICIT NON-LINEAR SCHEMES TO SEMI-IMPLICIT PROJECTION METHODS

STEFAN TUREK

Institute for Applied Mathematics, University of Heidelberg, INF 294, Heidelberg, Germany

SUMMARY

We present a numerical comparison of some time-stepping schemes for the discretization and solution of the non-stationary incompressible Navier–Stokes equations. The spatial discretization is by non-conforming quadrilateral finite elements which satisfy the LBB condition. The major focus is on the differences in accuracy and efficiency between the backward Euler, Crank–Nicolson and fractional-step θ schemes used in discretizing the momentum equations. Further, the differences between fully coupled solvers and operator-splitting techniques (projection methods) and the influence of the treatment of the nonlinear advection term are considered. The combination of both discrete projection schemes and *non-conforming finite elements* allows the comparison of schemes which are representative for many methods used in practice. On Cartesian grids this approach encompasses some well-known staggered grid finite difference discretizations too. The results which are obtained for several typical flow problems are thought to be representative and should be helpful for a fair rating of solution schemes, particularly in long-time simulations.

KEY WORDS: non-stationary incompressible Navier–Stokes equations; time-stepping schemes; projection methods

1. INTRODUCTION

We consider numerical solution techniques for the non-stationary incompressible Navier–Stokes equations

$$\mathbf{u}_t - \nu\Delta\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p = \mathbf{f} \quad \text{and} \quad \nabla\cdot\mathbf{u} = 0 \quad \text{in} \quad \Omega\times(0,T], \tag{1}$$

for given force $\mathbf{f}$ and viscosity ν, with prescribed boundary values on the boundary $\partial\Omega$ and an initial condition at $t=0$. Solving this problem numerically is still a considerable task in the case of long-time calculations and higher Reynolds numbers, particularly in 3D. In this paper we concentrate on the 2D case, which is representative also for 3D problems. Related results in 3D can be found in Reference 1.

Apart from rather 'exotic' schemes such as *discontinuous space–time Galerkin methods*[2] and *characteristic methods*,[3] the common solution approach is a separate discretization in space and time.

We first (semi)discretize in time by one of the usual methods known from the treatment of ordinary differential equations, such as the forward or backward Euler, Crank–Nicolson or fractional-step θ scheme,[4] to obtain a sequence of generalized stationary Navier–Stokes problems. *Given* $\mathbf{u}^n$ *and the time step* $k = t_{n+1} - t_n$, *then solve for* $\mathbf{u} = \mathbf{u}^{n+1}$ *and* $p = p^{n+1}$

$$\frac{\mathbf{u}-\mathbf{u}^n}{k} + \theta(-\nu\Delta\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u}) + \nabla p = \mathbf{g}^{n+1} \quad \text{and} \quad \nabla\cdot\mathbf{u} = 0 \quad \text{in} \quad \Omega, \tag{2}$$

CCC 0271–2091/96/100987–25

Received 10 March 1995
Revised 8 August 1995

with right-hand side

$$\mathbf{g}^{n+1} := \theta \mathbf{f}^{n+1} + (1-\theta)\mathbf{f}^n - (1-\theta)(-\nu \Delta \mathbf{u}^n + \mathbf{u}^n \cdot \nabla \mathbf{u}^n). \tag{3}$$

The parameter θ has to be chosen depending on the time-stepping scheme, e.g. $\theta = 1$ for the backward Euler scheme or $\theta = \frac{1}{2}$ for the Crank–Nicolson scheme. The pressure term $\nabla p = \nabla p^{n+1}$ may be replaced by

$$\theta \nabla p^{n+1} + (1-\theta)\nabla p^n, \tag{4}$$

but, with appropriate postprocessing, both strategies lead to solutions of the same accuracy. In all cases we end up with the task of solving in each time step a non-linear saddle point problem of type (2) which then has to be discretized in space.

The resulting non-linear systems of equations may be treated by a coupled approach in $\mathbf{u}$ and p,[5] which promises the best stability behaviour but also entails the largest numerical effort. Further variants, known as projection methods,[6] decouple pressure and velocity, which reduces the problem to the solution of a sequence of 'simple' (scalar) problems. However, at the same time, they lead to smaller time steps owing to the inherently more explicit character and often suffer from spurious pressure boundary layers. Finally, concerning the treatment of the non-linearity, we are left with the problem: should we solve 'exactly' the non-linear problems in each time step, or should we linearize the advection term, e.g. by extrapolation in time, or might an explicit treatment already give satisfactory results?

These different techniques lead to a large variety of schemes, all of which have been occurring in practice for years. Theoretical considerations can provide some ideas concerning stability of these schemes, convergence rates for subproblems, necessary time step sizes or qualitative behaviour for large Reynolds numbers, but a complete analysis or quantitative prediction is not possible today. Therefore the only way to make a judgement is to perform numerical tests, at least for some classes of problems which seem to be representative. However, looking into the literature, it seems that there have not been many studies of this type which can give satisfactory answers.

We suppose that one of the main reasons is the difficulty in finding a spatial discretization which allows for the design of efficient as well as robust methods for all types of schemes mentioned. Of course, for a fair comparison, all schemes should produce the 'same' results (up to round-off errors) if the time step is small enough and the elapsed computer time has to be acceptable on ordinary workstations. For higher-order finite elements, e.g. those using at least quadratic interpolation for the velocity, 'fast' projection methods can be implemented, but the realization of robust and efficient coupled solvers, e.g. of block Gauss–Seidel type,[7] is much harder. On the other hand, piecewise constant pressure approximations allow for very fast coupled multigrid techniques, but it is not so clear how to use them in projection schemes which involve the solution of discrete Poisson problems with these elements. Additionally, many well-known discretization schemes using continuous equal-order interpolation for velocity and pressure need additional stabilization which is difficult to optimize. For example, projection schemes implicitly introduce some stabilization terms,[8] but these may not be sufficient for small time steps. What we need (at least for our comparisons) is a (finite element) discretization scheme and a solution procedure satisfying the following.

1. The finite element spaces for $\mathbf{u}$ and p are stable, i.e. satisfy the LBB condition.[9]
2. A robust and efficient coupled solver is available.
3. A robust and efficient solver of projection type is available.
4. A non-linear solution strategy is available.

A method which seems to satisfy all these requirements consists of *discrete projection schemes* with non-conforming linear or *rotated multilinear* finite elements for $\mathbf{u}$ and piecewise constant approximations for p.[10,11] With this approach we can develop very efficient solution schemes of both coupled and

projection type, with a special non-linear or linearized treatment of the advection. The resulting solutions are coincident (as soon as the time steps are small enough) and no spurious pressure oscillations occur. This approach will be the basis of our theoretical and numerical investigations which examine the following aspects.

1. What is the influence of the type of discretization for the time derivative $\mathbf{u}_t$? We consider the backward Euler, Crank–Nicolson and fractional-step θ schemes. We restrict ourselves to implicit methods only (with $\theta \geqslant 0{\cdot}5$), at least for the diffusion part, since many tests show that otherwise the time steps have to be chosen too small to guarantee stability, and recently very efficient solution methods for this kind of implicit scheme have become available. The main focus will be on the difference between the Crank–Nicolson and fractional-step θ schemes, which are both of second-order accuracy and have the same numerical cost. Also, the fractional-step θ scheme has (theoretically) better stability properties as it stems from a strongly A-stable difference formula.[12]
2. What is the influence of the various ways of coupling the velocity $\mathbf{u}$ with the pressure p? Recently we have shown[11] that the class of discrete projection methods contains both variants, $\mathbf{u}$–p coupled solvers as well as operator-splitting schemes, and a parameter-controlled transition between the two is possible. We have shown by a complexity analysis that particularly in the case of higher Reynolds numbers the projection steps can be performed in a much more efficient way. However, the question which cannot be answered theoretically is: how much smaller must the time steps be chosen to guarantee results of the same quality as obtained by the fully coupled approach, and what about the total CPU time?
3. What is the influence of the treatment of the non-linear advection term? It is necessary to use a fully non-linear iteration in each time step or is it sufficient to linearize the advection by extrapolation in time? This means that we replace the non-linear term $\mathbf{u}^{n+1} \cdot \nabla \mathbf{u}^{n+1}$ in a semi-implicit way by

$$\mathbf{u}^n \cdot \nabla \mathbf{u}^{n+1} \quad \text{or} \quad (2\mathbf{u}^n - \mathbf{u}^{n-1}) \cdot \nabla \mathbf{u}^{n+1}. \tag{5}$$

Another possibility is a fully explicit treatment of this term as $\mathbf{u}^n \cdot \nabla \mathbf{u}^n$ whereby only symmetric linear systems have to be solved, but what about the necessary time steps?

For a given spatial mesh we always try to reach the same accuracy for all methods proposed, with different time steps which are chosen by hand or by an adaptive time step control. This control, which 'optimally' adapts the necessary time steps for each scheme, and the fact that we optimized all implemented versions in the same way should guarantee a fair comparison. In the end we obtain the total CPU time needed to reach a prescribed time level, which is then taken as the final measure for the quality of the various schemes.

2. SPATIAL AND TIME DISCRETIZATION

We first discretize the time derivative in the Navier–Stokes equations (1) by one of the usual time-stepping schemes, with prescribed boundary values for every time step. *Given* $\mathbf{u}^n$ *and the time step* $k = t_{n+1} - t_n$, *then solve for* $\mathbf{u} = \mathbf{u}^{n+1}$ *and* $p = p^{n+1}$

$$\frac{\mathbf{u} - \mathbf{u}^n}{k} + \theta(-\nu\Delta\mathbf{u} + \mathbf{u} \cdot \nabla\mathbf{u}) + \nabla p = \mathbf{g}^{n+1} \quad \text{and} \quad \nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \tag{6}$$

with right-hand side

$$\mathbf{g}^{n+1} := \theta\mathbf{f}^{n+1} + (1-\theta)\mathbf{f}^n - (1-\theta)(-\nu\Delta\mathbf{u}^n + \mathbf{u}^n \cdot \nabla\mathbf{u}^n). \tag{7}$$

In the past, explicit time-stepping schemes have been commonly used in non-stationary flow calculations, but because of the severe stability problems inherent in this approach, the required small time steps prohibited the long-time solution of really time-dependent flows. Owing to the high stiffness, one is limited to implicit schemes in the choice of time-stepping methods for solving this problem. Since implicit methods have become feasible thanks to more efficient linear solvers, the schemes most frequently used are either the simple first-order backward Euler scheme (BE) with $\theta = 1$ *or the second-order Crank–Nicolson scheme (CN) with* $\theta = \frac{1}{2}$. *These two methods belong to the group of* one-step θ *schemes*. The CN scheme occasionally suffers from unexpected instabilities because of its weak damping property (not strongly A-stable), while the BE scheme is of first-order accuracy only. Another method which seems to have the potential to excel in this competition is the *fractional-step* θ *scheme* (FS). It uses three different values for θ and for the time step k at each time level. For a realistic comparison we define a macro time step with $K = t_{n+1} - t_n$ as a sequence of three time steps of (possibly variable) size k. Then in the case of the backward Euler or Crank–Nicolson scheme we perform three substeps with the same θ as above and time step $k = K/3$.

For the fractional-step θ scheme we proceed as follows. Choosing $\theta = 1 - \sqrt{2}/2$, $\theta' = 1 - 2\theta$, $\alpha = (1 - 2\theta)/(1 - \theta)$ and $\beta = 1 - \alpha$, the macro time step $t_n \to t_{n+1} = t_n + K$ is split into three consecutive substeps (with $\tilde{\theta} := \alpha\theta K = \beta\theta' K$):

$$
\begin{aligned}
[I + \tilde{\theta}N(\mathbf{u}^{n+\theta})]\mathbf{u}^{n+\theta} + \theta K\nabla p^{n+\theta} &= [I - \beta\theta KN(\mathbf{u}^n)]\mathbf{u}^n + \theta K\mathbf{f}^n, \qquad \nabla\cdot\mathbf{u}^{n+\theta} = 0,\\
[I + \tilde{\theta}N(\mathbf{u}^{n+1-\theta})]\mathbf{u}^{n+1-\theta} + \theta' K\nabla p^{n+1-\theta} &= [I - \alpha\theta' KN(\mathbf{u}^{n+\theta})]\mathbf{u}^{n+\theta} + \theta' K\mathbf{f}^{n+1-\theta}, \qquad \nabla\cdot\mathbf{u}^{n+1-\theta} = 0,\\
[I + \tilde{\theta}N(\mathbf{u}^{n+1})]\mathbf{u}^{n+1} + \theta K\nabla p^{n+1} &= [I - \beta\theta KN(\mathbf{u}^{n+1-\theta})]\mathbf{u}^{n+1-\theta} + \theta K\mathbf{f}^{n+1-\theta}, \qquad \nabla\cdot\mathbf{u}^{n+1} = 0.
\end{aligned}
$$

Here and in the following we use the more compact form for the diffusive and advective parts:

$$N(\mathbf{u})\mathbf{u} := -\nu\Delta\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u}. \tag{8}$$

Being a strongly A-stable scheme, the FS method possesses the full smoothing property, which is important in the case of rough initial or boundary values. Further, it contains very little numerical dissipation, which is crucial in the computation of non-enforced temporal oscillations in the flow. A rigorous theoretical analysis of the FS scheme[12–14] applied to the Navier–Stokes problem establishes second-order accuracy for this special choice of θ. Therefore this scheme should combine the advantages of both the classical CN scheme (second-order accuracy) and the BE scheme (strongly A-stable), but with the same numerical effort.

Thus in each time step we have to solve non-linear problems of the following type. *Given* $\mathbf{u}^n$, *parameters* $k = k(t_{n+1})$, $\theta = \theta(t_{n+1})$ *and* $\theta_i = \theta_i(t_{n+1})$, $i = 1, \ldots, 3$, *then solve for* $\mathbf{u} = \mathbf{u}^{n+1}$ *and* $p = p^{n+1}$

$$[I + \theta kN(\mathbf{u})]\mathbf{u} + k\nabla p = [I - \theta_1 kN(\mathbf{u}^n)]\mathbf{u}^n + \theta_2 k\mathbf{f}^{n+1} + \theta_3 k\mathbf{f}^n, \qquad \nabla\cdot\mathbf{u} = 0. \tag{9}$$

For spatial discretization we choose a finite element approach. In setting up a finite element model of the Navier–Stokes equations, one starts with a variational formulation. On the finite mesh T_h (triangles, quadrilaterals or their analogues in 3D) covering the domain Ω with local element width h, one defines polynomial trial functions for velocity and pressure. These spaces $\mathbf{H}_h$ and $\mathbf{L}_h$ should lead to numerically stable approximations as $h \to 0$, i.e. they should satisfy the *Babuska–Brezzi condition* with a mesh-independent constant γ:

$$\min_{p_h \in L_h} \max_{\mathbf{v}_h \in \mathbf{H}_h} \frac{(p_h, \nabla\cdot\mathbf{v}_h)}{\|p_h\|_0 \|\nabla\mathbf{v}_h\|_0} \geqslant \gamma > 0. \tag{10}$$

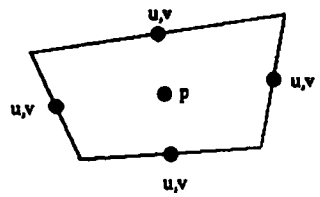

Figure 1. Nodal points of non-conforming finite element pair

Many stable pairs of finite element spaces have been proposed in the literature. Our favourite candidate is a quadrilateral element which (in 2D) uses piecewise *rotated bilinear* shape functions for the velocities, spanned by $\langle x^2 - y^2, x, y, 1\rangle$, and piecewise constant pressure approximations (see Figure 1). The nodal values are the mean values of the velocity vector over the element edges and the mean values of the pressure over the elements, rendering this approach 'non-conforming'. This element is the natural quadrilateral analogue of the well-known triangular Stokes element of Crouzeix–Raviart[15] and can also be defined in three space dimensions. A convergence analysis is given in Reference 10 and very promising computational results are reported in References 5, 16 and 17.

This element pair has several important features. It admits simple upwind strategies which lead to matrices with certain M-matrix properties. Further, efficient multigrid solvers are available which work satisfactorily over the whole range of relevant Reynolds numbers, $1 \leqslant Re \leqslant 10^5$, and also on non-uniform meshes. In Reference 11 we have even shown by a complexity analysis that this pair of elements is most efficient in the case of highly non-stationary flows. In combination with the *discrete projection methods* it works very robustly and efficiently in a multigrid code also on highly stretched and anisotropic grids.

Using the same symbols $\mathbf{u}$ and p also for the coefficient vectors in the nodal representation for the functions $\mathbf{u}$ and p, the discrete version of problem (9) may be written as a (non-linear) algebraic system of the following form. *Given* $\mathbf{u}^n$, *a right-hand side* $\mathbf{g}$ *and a time step* k, *then solve for* $\mathbf{u} = \mathbf{u}^{n+1}$ *and* $p = p^{n+1}$

$$S\mathbf{u} + kBp = \mathbf{g}, \qquad B^{\mathrm{T}}\mathbf{u} = 0, \tag{11}$$

with matrix S *and right-hand side* $\mathbf{g}$ *such that*

$$S\mathbf{u} = [M + \theta kN(\mathbf{u})]\mathbf{u}, \qquad \mathbf{g} = [M - \theta_1 kN(\mathbf{u}^n)]\mathbf{u}^n + \theta_2 k\mathbf{f}^{n+1} + \theta_3 k\mathbf{f}^n. \tag{12}$$

Here M is the *mass* matrix and $N(.)$ the *advection* matrix containing the diffusive and convective parts corresponding to the non-linear form in (8). For dominant transport the advection part may include some stabilization, e.g. some upwind mechanism.[5] B is the *gradient* matrix and $-B^{\mathrm{T}}$ the transposed *divergence* matrix. With M_{l} we denote the lumped mass matrix, which is a diagonal matrix.

3. SOLUTION TECHNIQUES FOR DISCRETE EQUATIONS

In the last section we arrived at the following discrete non-linear problems for $\mathbf{u}$ and p to be solved in each time step. *Given a right-hand side* $\mathbf{g}$ *with appropriate boundary conditions, then solve*

$$S\mathbf{u} + kBp = \mathbf{g}, \qquad B^{\mathrm{T}}\mathbf{u} = 0. \tag{13}$$

Two possible approaches are as follows.

1. We first treat the non-linearity by an outer non-linear iteration of fixed point or quasi-Newton type or by a linearization technique through extrapolation in time to obtain linear indefinite subproblems which can be solved by a coupled or splitting approach.

2. We first split the coupled problem to obtain definite problems in $\mathbf{u}$ (Burgers equations) as well as in p (linear pressure Poisson problems). Then we treat the non-linear problems in $\mathbf{u}$ by an appropriate non-linear iteration or by a linearization technique.

In our applications the non-linear problems are solved by the *adaptive fixed point defect correction method*,[5] while for the solution of the coupled problems or for their decoupling the *discrete projection method* formalism is used.[11]

3.1. Treatment of the non-linear problems

We perform a number of non-linear iteration steps by the *adaptive fixed point defect correction method*. In each iteration an auxiliary linear problem corresponding to the Frechét derivative of the Navier–Stokes operator (or Burgers operator) has to be solved. The 'exact' derivative leads to matrices including 'convection' terms $\mathbf{u}^n \cdot \nabla \mathbf{u}^{n+1}$ which can be stabilized by upwinding, but also 'reaction' terms of the form $\mathbf{u}^{n+1} \cdot \nabla \mathbf{u}^n$ which may cause instabilities for larger time steps. We prefer to simply delete the 'bad' reaction terms, leading to an approximate Frechét derivative only. We explain this procedure in detail for the abstract non-linear equation

$$T(u)u = f, \tag{14}$$

where $T(.)$ represents the Navier–Stokes or the Burgers operator. On the discrete level our procedure is equivalent to solving in each non-linear iteration $\mathbf{l}$ as Oseen or a convection–diffusion equation in $\mathbf{u}_l$ (and p_l):

$$u_l = u_{l-1} - \omega_l [\tilde{T}(u_{l-1})]^{-1} [T(u_{l-1})u_{l-1} - f], \tag{15}$$

with $\omega_l > 0$ being chosen adaptively. The operator $\tilde{T}(u_{l-1})$ represents the approximate Frechét derivative and may be chosen as $\tilde{T}(u_{l-1}) = T(u_{l-1})$. However, other choices are possible (see below). This non-linear iteration may be terminated if the residual $T(u_{l-1})u_{l-1} - f$ is small enough or if l exceeds a given limit N.

In the non-stationary case we can also stop with $N = 1$ (and $\omega_l = 1$). This approach can be interpreted as a linearization of the convective term by a constant extrapolation in time, e.g. replacing

$$\mathbf{u}^{n+1} \cdot \nabla \mathbf{u}^{n+1} \quad \text{by} \quad \mathbf{u}^n \cdot \nabla \mathbf{u}^{n+1}. \tag{16}$$

The corresponding time-stepping scheme is of first order only. A simple remedy for obtaining second-order accuracy with about the same numerical cost is to use a linear extrapolation in time, i.e. using

$$(2\mathbf{u}^n - \mathbf{u}^{n-1}) \cdot \nabla \mathbf{u}^{n+1} \quad \text{as approximation for} \quad \mathbf{u}^{n+1} \cdot \nabla \mathbf{u}^{n+1}. \tag{17}$$

For both schemes we have to build up the system matrices in each time step. The corresponding linear systems are non-symmetric and require special solution techniques. A much simpler and very common possibility is a fully explicit treatment of the non-linearity, i.e. replacing

$$\mathbf{u}^{n+1} \cdot \nabla \mathbf{u}^{n+1} \quad \text{by} \quad \mathbf{u}^n \cdot \nabla \mathbf{u}^n \tag{18}$$

and considering this advection term as part of the right-hand side. The resulting linear systems are symmetric quasi-Stokes or positive definite quasi-Poisson equations.

It is obvious that the fully non-linear iteration schemes carry more numerical cost for each time step than the linearization techniques (with only one iteration). Additionally, the fully explicit treatment consumes the least effort. However, the question not included in these considerations is that concerning the necessary size of the time step. The fully non-linear iteration is expected to be more stable and accurate, while the price to be paid for the more explicit schemes is a smaller time step size. Thus, asking for the total CPU time, which schemes are the more efficient ones?

3.2. *Treatment of coupled equations*

We treat the coupled problems of type (13), which may be linear or non-linear, by the *discrete projection method*, which reads (with $L \geqslant 1$, $\alpha > 0$, mostly $\alpha = 1, 2$) as follows.

General non-linear version of Discrete Projection Method (L). Given $\mathbf{u}^0 := \mathbf{u}_0$ *and* $p^0 := p_0$, *then solve for* $l = 1, \ldots, L$

$$p^l = p^{l-1} + \frac{\alpha}{k}(B^{\mathrm{T}}C^{-1}B)^{-1}B^{\mathrm{T}}S^{-1}(\mathbf{g} - kBp^{l-1}). \tag{19}$$

Finally, set $p := p^L$, $\tilde{\mathbf{u}} = S^{-1}(\mathbf{g} - kBp^{L-1})$ and determine $\mathbf{u}$ via

$$\mathbf{u} = \tilde{\mathbf{u}} - \frac{k}{\alpha}C^{-1}B(p^L - p^{L-1}). \tag{20}$$

If the matrix S corresponds to a discretized linear Oseen or quasi-Stokes operator, we can rewrite this algorithm in the following way.

Linear version of Discrete Projection Method (L). Given $\mathbf{u}^0 := \mathbf{u}^0$ and $p^0 := p_0$, then solve for $l = 1, \ldots, L$

$$p^l = p^{l-1} - \alpha(B^{\mathrm{T}}C^{-1}B)^{-1}\left(B^{\mathrm{T}}S^{-1}Bp^{l-1} - \frac{1}{k}B^{\mathrm{T}}S^{-1}\mathbf{g}\right). \tag{21}$$

Finally, set $p := p^L$ *and calculate* $\mathbf{u}$ *via*

$$S\mathbf{u} = \mathbf{g} - kBp + \frac{k}{\alpha}(\alpha I - SC^{-1})B(p^L - p^{L-1}). \tag{22}$$

This scheme is nothing else but the classical α-relaxed Richardson iteration for the Schur complement form of problem (13) with the special preconditioner $B^{\mathrm{T}}C^{-1}B$. The operator C should be a good approximation to S, but at the same time C^{-1} should be easily applied. The addictive term $(k/\alpha)(\alpha I - SC^{-1})B(p^L - p^{L-1})$ is introduced to ensure that $\mathbf{u}$ is *discretely divergence-free*, i.e.

$$B^{\mathrm{T}}\mathbf{u} = 0, \tag{23}$$

and no additional evaluation of S^{-1} is needed in this step. If S is the non-linear operator, we use for its 'inversion' the proposed *adaptive fixed point defect correction method.*

There are various possibilities for choosing the preconditioner C and the iteration limit L. In Reference 20 we have shown that there are at least two promising approaches.

1. First we use the *adaptive fixed point defect correction method* as an outer iteration. Then we have to solve linear coupled equations of quasi-Stokes or Oseen type in each iteration. If we choose $C = S$, we have to realize the application of $(B^{\mathrm{T}}S^{-1}\mathrm{B})^{-1}$ by performing directly a solution step with the linear indefinite system

$$A = \begin{bmatrix} S & kB \\ B^{\mathrm{T}} & 0 \end{bmatrix}. \tag{24}$$

A good solver for these linear equations is usually obtained by a direct multigrid approach. Owing to the definitions of A, certain block Gauss–Seidel or distributive smoothers[7,18] may successfully be used. However, the numerical effort for each iteration is large and severe problems still exist on highly anisotropic grids.[19] There is some work going on to develop block ILU or other linewise working methods, but especially the efficient and robust solution of the 3D problem on general grids is still an open problem. Additionally, we have seen in Reference 11 that, assuming large Reynolds numbers (and

correspondingly small time steps k), S can be interpreted as a non-symmetric but well-conditioned perturbation of the mass matrix M. On the other hand, we have for the equivalent Schur complement equation, and equivalently for the matrix A in the case of small k,

$$\text{cond}(B^T S^{-1} B) = \text{cond}(B^T [M + O(k)]^{-1} B) \approx \text{cond}(\Delta_h) = O(h^{-2}), \tag{25}$$

with Δ_h being a discrete Laplacian. This means that for small time steps the condition number of the coupled system is bounded below by $O(h^{-2})$, resulting from a second-order problem due to the incompressibility constraint, and no further improvement may be gained even for smaller k. Hence we see that the coupled matrix A in (24) consists of a time-dependent large part S acting on the velocity components, which gets 'better' for small k, and a much smaller 'projection' part $B^T S^{-1} B \approx B^T M^{-1} B \approx \Delta_h$ acting on the pressure, which is (almost) time-step-invariant. However, this small part determines the overall convergence rate of the coupled solvers for small k.

Therefore, for really non-stationary problems requiring small time steps, the efficiency of this coupled technique does not improve as expected. This is due to the central idea of using efficient Stokes solvers as basic components even for the fully non-stationary non-linear Navier–Stokes equations. A completely different approach is to choose the matrix $C = M_l$ for preconditioning. This is an 'exact' preconditioner for the case of divergence-free L^2-projection and can be expected to work well for small k only, as for low Reynolds numbers and large time steps the corresponding convergence rates may deteriorate. Other diagonal matrices as choices for C (which is similar to SIMPLE-type methods; for an overview see Reference 20) and the distributive smoothers of Wittum[18] are possible candidates too. For really non-stationary flows these approaches seem to be more appropriate than the fully coupled techniques.

We have seen in Reference 11 that especially for the non-conforming finite elements used, the application of S^{-1} and $(B^T M_l^{-1} B)^{-1}$ leads to very robust and efficient solvers involving relatively low numerical cost. Each iteration step may be expected to be 'faster' by at least a factor of nine (in 2D) or 19 (in 3D) compared with the coupled method.[11] Our approach is to replace the approximate Frechét derivative in (15) by another approximation, namely by $L \geqslant 1$ steps of the (linear) discrete projection scheme. Our numerical tests even show that in most cases it is sufficient to perform $L = 1$ iteration with only one multigrid sweep for the resulting linear operators S and $P = B^T M_l^{-1} B$. Hence, in comparison with the coupled solvers, the total number of non-linear steps may increase, but at the same time in each non-linear iteration the numerical effort for solving the linear subproblems is drastically decreased.

2. The other possibility is to use directly the non-linear scheme (19), (20) for the solution of problem (13). This iteration can be viewed as a special kind of non-linear iteration scheme for solving the generalized stationary Navier–Stokes problem (13). First the velocity $\mathbf{u}$ and pressure p are decoupled in an outer iteration, while the non-linear transport–diffusion equation is solved in an inner iteration. However, we perform only $L = 1$ iteration. Nevertheless, it is obvious that this scheme works well for small time steps. This approach is well known and similar to a certain class of well-established iteration schemes, namely the 'classical' projection schemes.

Since the matrix $B^T M_l^{-1} B$ resembles a mixed discretization of the Laplacian operator,[6] this method is a discrete analogue of the schemes proposed by Chorin[21] (if we set $p_0 = 0$) and Van Kan[22] (if we choose $p_0 = p(t_n)$). The main difference is that we first discretize the equations as shown in (11) and then perform the projection steps (19) and (20). Hence our treatment of boundary values for the pressure p, which is one of the main problems with the continuous projection schemes, may be different. Taking the same (Dirichlet) boundary conditions in assembling the matrix M_l as for the complete matrix S, our numerical tests show good results (no significant boundary layers for higher Reynolds numbers). However, there are some other possibilities on the discrete level which may lead to different results. A more detailed analysis for these cases is in preparation and will be published in a forthcoming paper.

The different schemes we used are the following.

Coupled solution by Coupled solver: CC-{n,l,c,x}. We use the *adaptive fixed point defect correction method* as outer iteration, while for the linear coupled subproblems we choose $C = S$ and $\alpha = 1$. Hence the linear equations in the non-linear process are solved in one iteration step, meaning that $L = 1$. This solver is a multilevel-based approach[16] with a block Gauss–Seidel scheme as smoothing operation. If we perform $N \geqslant 1$ non-linear iterations until the residual vanishes, we denote this fully implicit coupled scheme by CC-n. If we choose $N = 1$, meaning only one non-linear iteration, we obtain the versions with time-extrapolated advection, namely CC-l (linear extrapolation) and CC-c (constant extrapolation). We omit the possibility of treating the non-linearity explicitly (version CC-x), since the resulting subproblems are of quasi-Stokes type and no gain in efficiency could be observed for our solution approach.

Coupled solution by Projection solver: CP-{n,l,c,x}. We use again the *adaptive fixed point defect correction method* as outer iteration and select the preconditioner $C = M_l$. We first consider the fully non-linear iteration, which is denoted by CP-n. If L is large enough, i.e. if the linear problems are solved accurately, then the same approximate Frechét derivative as in CC-n is taken in each non-linear step and the number of non-linear iterations must be the same as in the fully coupled version CC-n. However, we have seen that the case $L = 1$ leads to a different approximate Frechét derivative and therefore the number of non-linear steps will change. We usually perform $N = 9$ non-linear steps at the maximum. If we can solve the non-linear problem in less than nine steps, we get again the same solution as obtained by CC-n and the time step size is the same. If $N = 9$ is not sufficient, the resulting solution may be different and the time step size has to be reduced.

Performing only one non-linear iteration, we obtain the versions CP-c and CP-l. We solve the resulting linear problems again by L steps of our discrete projection scheme, with $L = 9$ iterations at the maximum. If $L = 9$ is not large enough, we arrive at a different solution as obtained by the coupled solvers CC-{l,c} and the time step size again has to be reduced.

Projection solution by Projection solver: PP-{n,l,c,x}. First we apply a decoupling step for $\mathbf{u}$ and p as outer iteration. We choose again $C = M_l$, but perform only $L = 1$ iteration with $\alpha = 2$ in each time step, which is equivalent to (19) and (20). In the fully non-linear case, PP-n, we use the same fixed point iteration as explained above for the transport–diffusion step with non-linear operator S. In an analogous way we treat (by extrapolation) the linearized schemes PP-c and PP-l and additionally we also test the explicit treatment of the non-linearity, PP-x.

All versions of CC-{n,l,c,x} and CP-{n,l,c,x} (and even PP-{n,l,c,x} for L large enough) lead to the 'same' solutions if the numbers of non-linear steps, N, and discrete projection steps, L, are large enough. The variants PP-{n,l,c,x} with $L = 1$ may lead to different results, but for time steps k chosen small enough the same solutions are obtained. The complexity analysis in Reference 11 shows that in 2D a single iteration of the coupled scheme ($C = S$) can be expected to cost about 10 times more than one iteration of the operator-splitting method ($C = M_l$). However, at the same time we have to use more non-linear sweeps or smaller time steps for $C = M_l$ owing to the more explicit character of the scheme. If we want to determine the numerical cost of all schemes proposed, we have to ask for the necessary time step size to achieve (approximately) the same solution quality. These estimates cannot be given by theoretical considerations but they have to be obtained by numerical tests.

Adaptive time step control. We apply two kinds of time step selection: by 'hand' and by an adaptive error control. The former is easy to explain. We perform tests with a sequence of fixed time steps until no significant differences can be detected in comparison with a reference solution. This 'exact' solution is calculated with a very small time step but on the same spatial mesh. A more practical approach is the following one which is based on the estimation of the truncation error.

We assume that the method used is second-order-accurate in time (an analogous procedure can be performed for the first-order schemes). We are at time level t_n and we want to calculate a solution at $t_{n+1} = t_n + k$. We denote by v_k the solution pair $\{\mathbf{u}, p\}$ which is obtained by time step size k. Let v denote the exact solution at t_{n+1}. Our aim is to find an appropriate value for k such that the following estimate holds for the relative error measured in a properly chosen norm $\|.\|$:

$$\|v - v_k\| \leqslant \varepsilon \|v\|. \tag{26}$$

Here we implicitly make the (heuristic) assumption that the error at starting level t_n is zero. In our calculations we mostly prescribe the tolerance parameter $\varepsilon = 10^{-3}$. Further, we assume the asymptotic error expansion

$$v - v_k \sim k^2 e(v) + O(k^4), \tag{27}$$

with an error term $e(v)$ which is independent of the time step. We perform two calculations for step sizes $\tilde{k}$ and $3\tilde{k}$, which means that we apply three substeps with $\tilde{k}$ and one step with $3\tilde{k}$. In the case of the *fractional-step θ scheme* we compare the three substeps with one Crank–Nicolson step with step size $3\tilde{k}$ (which has a slightly different constant in the error expansion). This approach cannot be verified by rigorous mathematical arguments, but the performed test calculations always worked very well. Next we define the value for the relative changes:

$$REL_{\tilde{k}} := \frac{\|v_{3\tilde{k}} - v_{\tilde{k}}\|}{\|v_{\tilde{k}}\|}. \tag{28}$$

Then by a linear combination of relation (27) for $\tilde{k}$ and $3\tilde{k}$ we obtain

$$e(v) \sim \frac{v_{\tilde{k}} - v_{3\tilde{k}}}{8\tilde{k}^2} \tag{29}$$

and hence

$$\frac{\|v - v_k\|}{\|v\|} \sim \frac{k^2}{8\tilde{k}^2} \frac{\|v_{3\tilde{k}} - v_{\tilde{k}}\|}{\|v_{\tilde{k}}\|} = \frac{k^2}{8\tilde{k}^2} REL_{\tilde{k}}. \tag{30}$$

This last relation leads to the following estimate for k when the relative error is bounded by the given tolerance ε as demanded in (26):

$$k^2 \leqslant 8\varepsilon\tilde{k}^2 \frac{1}{REL_{\tilde{k}}}. \tag{31}$$

Our strategy is as follows. Given a step size $\tilde{k}$, we perform three (sub)steps with a parameter $\tilde{k}$ and one step with $3\tilde{k}$. Afterwards we calculate the relative changes $REL_{\tilde{k}}$ and use this parameter to compute the necessary time step k such that the error in equation (26) is controlled by ε. If the estimated value k is much smaller than the time step $\tilde{k}$ used for its prediction, we repeat the last calculation with $\tilde{k} = k$. If the value for k is larger than the used one or only slightly smaller (say less than 50 per cent), we accept the result and perform the next macro time step, now with k and $3k$. Finally we obtain a higher-order accuracy if we perform an additional linear extrapolation step of the partial solution $v_{\tilde{k}}$ and $v_{3\tilde{k}}$.

This time step control is well known in the field of ordinary differential equations and works well in most calculations. The basic assumption is that we are already in the asymptotic range such that the error presentation in (27) is true, which, however, cannot be guaranteed in general. A second conceptual problem is that we can only give estimates for the local discretization error, hereby assuming 'exact' starting values for every time step. Although most of our numerical tests gave satisfactory results, there may occur problems, particularly in strongly non-stationary cases. Therefore we are planning in the future to incorporate a global residual-based error control as proposed in Reference 23.

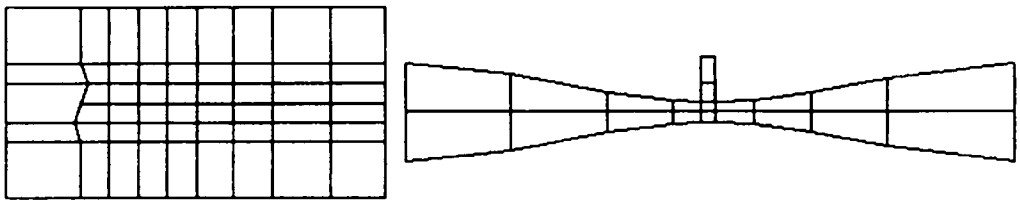

Figure 2. Coarse grids

4. NUMERICAL COMPARISONS

The two test problems we consider (see References 5 and 24 for more details) are

(i) von Kármán vortex shedding behind an inclined plate in a channel
(ii) flow in a Venturi pipe (a dynamic water pump device in a sailing boat).

Figure 2 shows the meshes used, which are refined systematically by connecting opposite midpoints. The resulting meshes in our multigrid calculations for the channel flow range from 13,500 elements (coarse level) to 54,000 elements (fine level) and for the Venturi pipe from 20,500 elements (coarse level) to 82,000 elements (fine level).

(i) The total length of the channel is $L_t = 10$, the height is $H_t = 5$ and the length of the plate is $L_p = 1$. At the inlet we prescribe a parabolic velocity profile with $U_{max} = 1$, while at the outlet a Neumann-type outflow condition is used.[24] The prescribed viscosity is $\nu = 1/500$. Figure 3 shows a typical snapshot of the (relative) streamlines for this problem and the oscillating (long-) time behaviour (up to $T = 60$, starting with Stokes flow) for the mean pressure difference across the plate defined by

$$P_{diff} := \oint_{plate_{left}} p \, ds - \oint_{plate_{right}} p \, ds. \tag{32}$$

The reason for choosing this quantity in our tests is the expected appearance of pressure boundary layers. Since P_{diff} is an important physical quantity, a 'good' method should be able to give accurate results for this quantity.

(ii) The total length of the Venturi pipe is $L_t = 32$, the height at the inlet is $H_t = 5$, the height in the interior is $H_i = 1$ and the width of the small upper channel is $W_i = 0{\cdot}8$. At the upper small 'inlet' and the right 'outlet' we prescribe the zero-mean-pressure condition,[24] while at the left inlet a parabolic velocity profile with $U_{max} = 1$ is prescribed, leading to maximum velocities of about 7 in the interior. At the narrowing a lower pressure is generated which enforces an incoming flux from the upper inlet, at least for the viscosity parameter $\nu = 1/1000$ used. Figure 4 shows a typical snapshot of the streamfunction and pressure for this problem and the (long-) time behaviour (up to $T = 30$) of the pressure and first velocity component at a certain boundary point in the right upper half of the domain.

All calculations are performed in FORTRAN 77 on Indigo (Silicon Graphics) or SS10 (SUN) workstations, which have about the same performance rating.

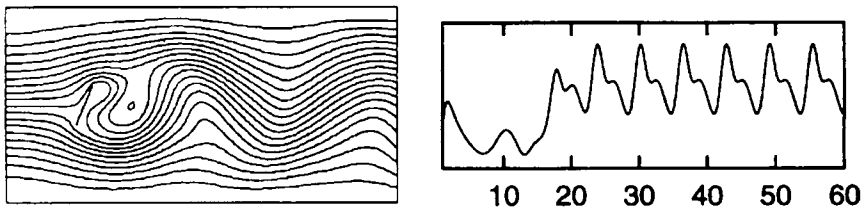

Figure 3. Streamline snapshot and mean pressure difference for channel flow

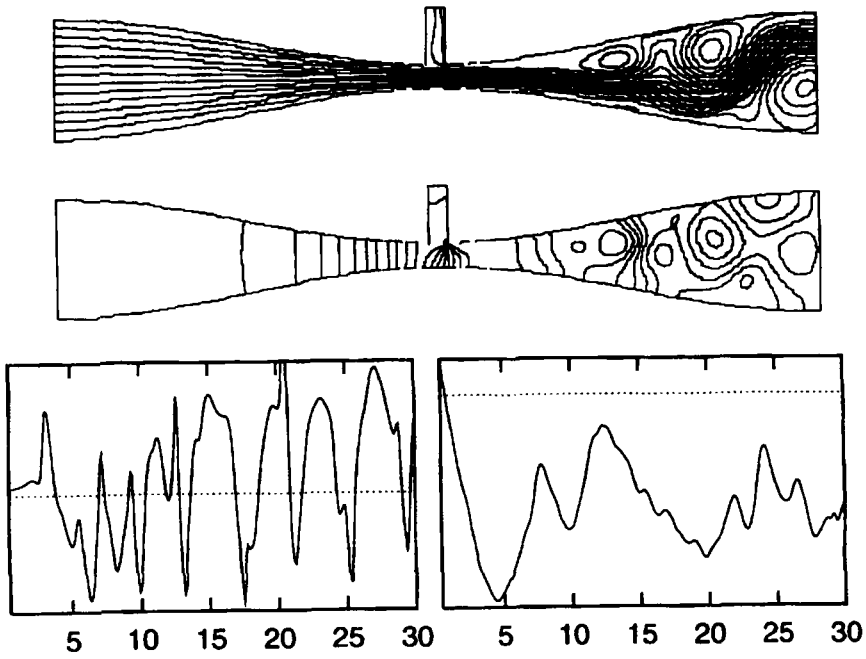

Figure 4. Streamline/pressure plot and velocity/pressure behaviour for Venturi pipe

4.1. Differences between first- and second-order schemes

We first concentrate on the results for the first-order schemes, namely the backward Euler scheme (BE) for the momentum equations and the constant extrapolation technique for the advection term ('c' versions). We show that both methods usually lead to results comparable with those obtained by the schemes of second order if the time step is drastically reduced (in our tests by at least a factor of 10). Further, no essential speed-up in efficiency can be found: the evaluation of the right-hand side in the Euler scheme can be slightly accelerated, but this is not essential in implicit methods.

The non-linear iteration can be avoided through simple constant extrapolation, but the same is true for the linearly extrapolated scheme, which is more accurate. Before we show some results, let us make a final remark concerning the first-order Chorin scheme.[21] It is straightforward to develop a corresponding discrete version in our context (setting $p_0 = 0$). The test calculations show that this method is too inaccurate and inefficient. In Figure 5 we show (relative) streamlines for the channel flow at $T = 10$ computed with the fully non-linear coupled versions (CC-n) of the CN and FS schemes (with $k = 0{\cdot}33$) and with the BE method for $k = 0{\cdot}33$ and $0{\cdot}033$. Here and in all other calculations the corresponding time step for the BE scheme has to be chosen smaller by at least a factor of 10. Figure 6 shows the time behaviour of the flux through the upper inlet in the Venturi pipe. We show corresponding results for CC-n via the FS and BE schemes, compared with the reference solution. Again the BE method forces us to choose at least $k = 0{\cdot}011$ to obtain similar results.

Similar results can be obtained for the 'c' versions, i.e. with constant extrapolation backwards in time for the advection term instead of using the fully non-linear iteration (Figures 7 and 8). Our first consequence is that we drop the BE scheme, the constant extrapolation techniques ('c' versions) and the Chorin scheme from the list of candidates as 'good' methods. Their total efficiency is much lower compared with the higher-order methods.

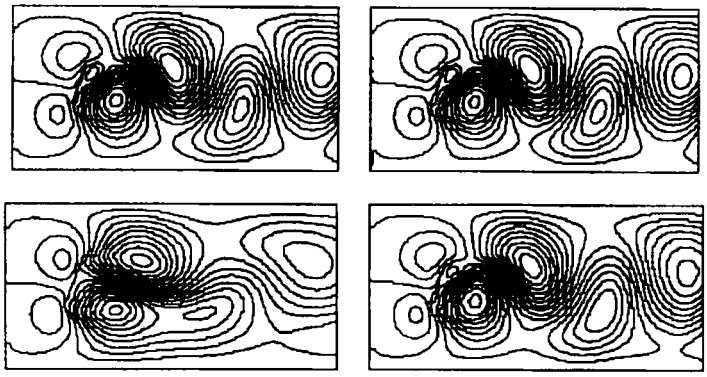

Figure 5. Relative streamlines of vortex shedding for channel flow: top, CN ($k = 0{\cdot}33$, left) and FS ($k = 0{\cdot}33$, right); bottom, BE ($k = 0{\cdot}33$, left) and BE ($k = 0{\cdot}033$, right)

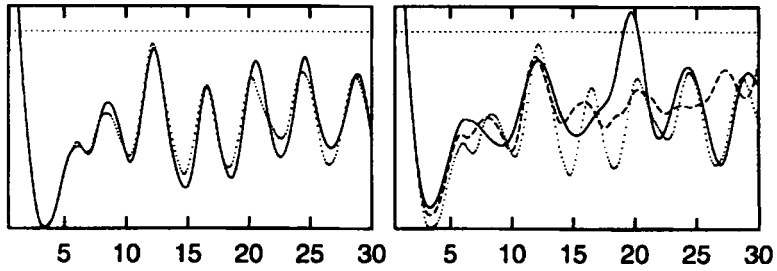

Figure 6. Total flux through upper inlet, calculated with FS (left) for $k = 0{\cdot}11$ (————) and BE (right) for $k = 0{\cdot}11$ (————) and 0·05 (– – – –), compared with reference solution (·····)

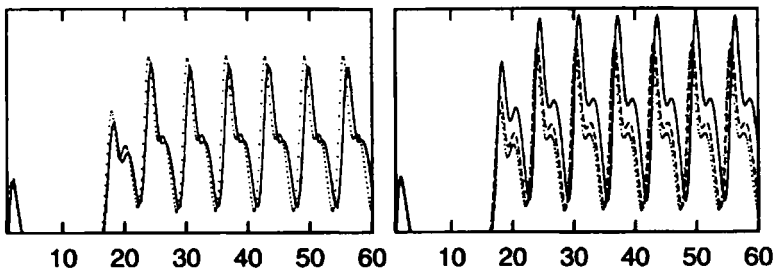

Figure 7. Mean pressure drop for channel flow, calculated via CC-n–CN (left, $k = 0{\cdot}33$, ————) and CC-c–CN (right, $k = 0{\cdot}11$, ————; $k = 0{\cdot}033$, (– – – –), compared with reference solution (·····)

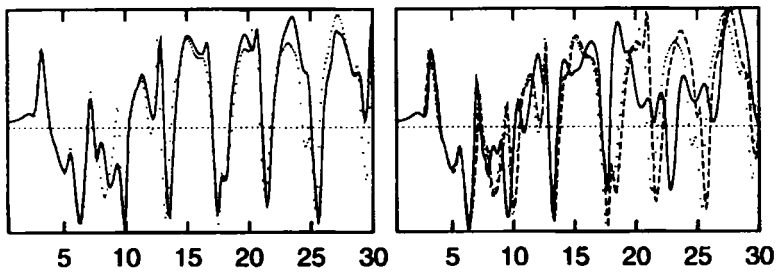

Figure 8. First velocity component at a single point for Venturi problem, calculated via CC-n–FS (left) with $k = 0{\cdot}11$ (————) and CC-c–FS (right) with $k = 0{\cdot}033$ (————) and 0·011 (– – – –), compared with reference solution (·····)

4.2. Differences between Crank–Nicolson and Fractional step θ schemes

As mentioned before, some more theoretical and numerical details about these schemes can be found in Reference 12. The essential theoretical results are as follows. Both are of second-order accuracy and lead to comparable numerical cost. However, the CN scheme is only A-stable while the FS method is strongly A-stable. This means that for rough initial values or boundary conditions the classical CN scheme may lead to numerical oscillations which are damped for smaller time steps only. We confirm these theoretical considerations numerically. The result is that for time steps k small enough, both schemes lead to approximate solutions with no significant differences, even for long-time calculations and high Reynolds numbers. However, if the time step is too coarse, the CN scheme tends to produce unphysical oscillations, which means that even for moderate time steps the CN scheme may be less robust and accurate compared with the FS scheme. By applying the adaptive time step control, these oscillations are damped, but generally a smaller time step has to be chosen to ensure more robustness.

Figures 9 and 10 show respectively the time behaviour of the mean pressure drop across the plate that of the total flux through the upper inlet, compared with reference solutions on the same mesh. Both calculations are performed with the schemes CC-n.

These results confirm that both schemes have about the same accuracy for realistic step sizes. We now consider their stability properties. As mentioned before, we expect problems for the CN scheme in coping with high-frequency perturbations caused by rough data. First, Figure 11 shows an instability effect of the CN scheme in the computation of the flow around a plate. Here the mean pressure drop shows non-physical fluctuations for $k = 0{\cdot}11$ which disappear for $k = 0{\cdot}05$. For the same step size the FS scheme is stable. For even coarser step sizes k, both schemes exhibit unstable behaviour. These results are obtained by the fully coupled solver CC-l with linear extrapolation of the advection direction, but the same behaviour could be obtained with the pure projection schemes PP. Additionally, our results are not limited to the case of pressure values on the boundary; the same holds also for points in the interior, i.e. all numerical oscillations are global effects.

Figure 12 demonstrates the last assertion: even the flux (measured through the upper inlet in the Venturi pipe) exhibits an unstable behaviour if calculated by the CN scheme.

Finally we show another deficiency of the CN scheme (Figure 13). For $k = 0{\cdot}033$ (which seems to be almost small enough) it produces a non-physical solution with positive outflow through the upper limit inlet (i.e. the boat might sink!).

These results show that the classical CN scheme and the FS scheme are essentially of the same accuracy and efficiency as soon as the time step is small enough. Nevertheless, there may occur problems for the CN scheme, due to the loss of strong A-stability, which may produce unphysical fluctuations of the solution. Therefore we also drop the Crank–Nicolson scheme in the following calculations and confine ourselves to the FS scheme.

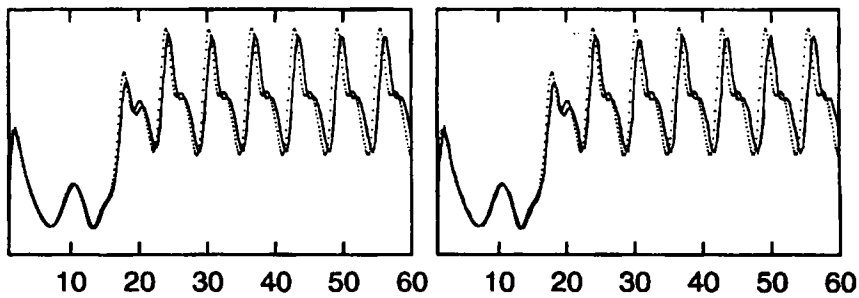

Figure 9. Mean pressure drop for channel flow, calculated by CC-n–CN (left) and CC-n–FS (right) with $k = 0{\cdot}33$ (————), compared with reference solution (·····)

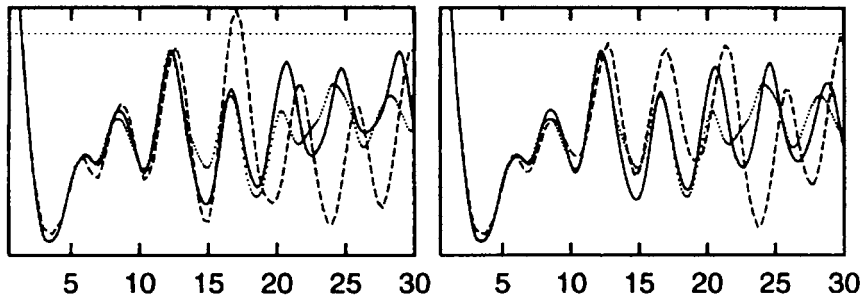

Figure 10. Flux for Venturi pipe, calculated by CC-n–CN (left) and CC-n–FS (right) with $k = 0{\cdot}33$ (– – – –) and 0·11 (————), compared with reference solution (·····)

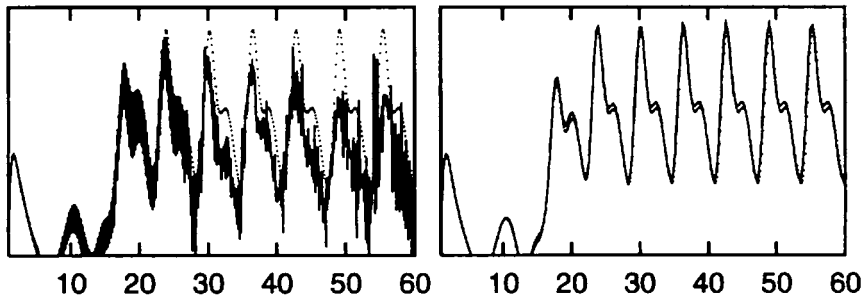

Figure 11. Mean pressure drop for channel flow, calculated by CC-l–CN (left) and CC-l–FS (right) with $k = 0{\cdot}11$ (————), compared with reference solution (·····)

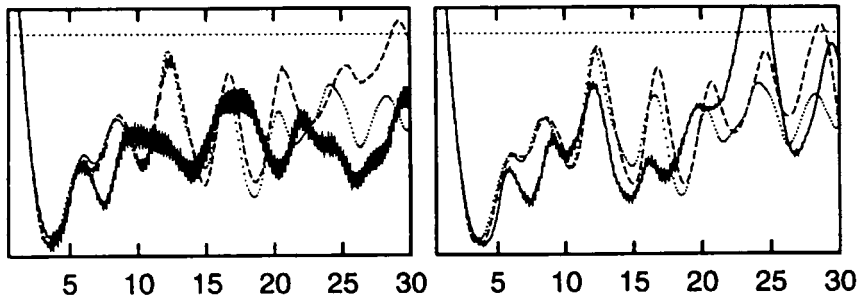

Figure 12. Total flux for Venturi pipe, calculated by CC-l–CN (left) and CC-l–FS (right) with $k = 0{\cdot}11$ (————) and 0·033 (– – – –), compared with reference solution (·····)

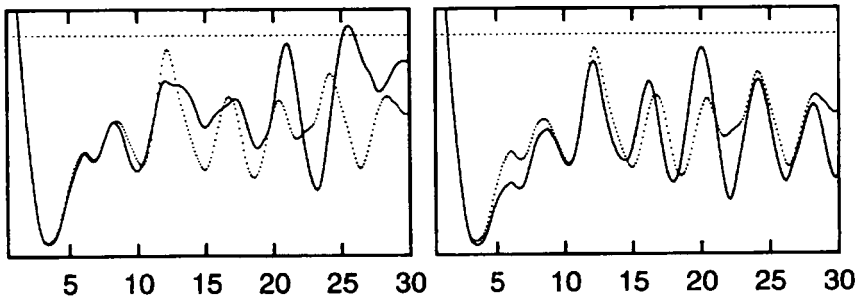

Figure 13. Total flux for Venturi pipe, calculated by PP-n–CN (left) and PP-n–FS (right) with $k = 0{\cdot}033$ (————), compared with reference solution (·····)

4.3. Differences between coupled and operator-splitting schemes

We now examine the differences between the proposed methods CC (coupled solution by coupled solver), CP (coupled solution by projection solver, $L \geqslant 1$) and the special case PP (projection solution by projection solver, with $L = 1$). In this comparison we always examine the non-linear versions; the other possibilities are discussed later.

Owing to the more explicit character of the projection scheme ($L = 1$), we expect some problems with the robustness if the step size is too large. In contrast, the fully implicit coupled solvers show a much better stability behaviour. Figure 14 shows the second velocity component for the channel flow. Even for very large time steps we detect vortex shedding behind the plate. This fact could be easily used for a demonstration of the 'perfect' speed of the code, since, even for very large time steps, video films with periodic vortex shedding can be generated. They look 'physical', but the solutions are inaccurate, particularly with respect to the frequency of the oscillations.

In contrast, the projection schemes tend to exhibit numerical oscillations as soon as a critical limit for k is exceeded; see Figure 15. However, this defect can be suppressed by the use of an adaptive time step control; without this control mechanism, projection schemes may be worthless. The same phenomena can be observed for the Venturi pipe problem. However, for large time steps k the solutions obtained by CC-n seem to be too inaccurate and therefore worthless (see Figure 16).

We have seen that stability requirements force us to use smaller step sizes k with the projection schemes PP. The same is true if the proposed schemes are used to calculate steady state solutions (for low Reynolds numbers). In this case the time step size is restricted by stability requirements only, since the computed stationary solutions are always the same. The necessary time steps for the channel flow with $\nu = \frac{1}{2}$ are determined by the adaptive time step control. The versions CP differ from CC as described in the previous section, since only a fixed number of non-linear iteration steps are performed. For comparison we show the corresponding results when a stationary version of the scheme CC-n is used.[5] The suffixes 'c' and 'f' in Table I indicate that the calculations are performed on the coarse and the fine mesh respectively. It is obvious that for low Reynolds numbers and particularly for stationary solutions the coupled solution techniques are favourable, at least if the mesh is not too anisotropic: the stationary solver is much faster. However, we are much more interested in the case for large Reynolds numbers when the flow becomes non-stationary. Then theoretical considerations in Reference 11 show that the projection techniques improve and for highly non-stationary flow they become superior. However, the 'practical' question is: what time step has to be used to achieve the same accuracy?

Tables II–V show the results for higher Reynolds numbers, again for the channel flow and the Venturi pipe problem. The surprising fact is that the time steps required to obtain the same high accuracy as observed by the coupled methods are only moderately smaller (a factor of 1·5–5). Since the expected

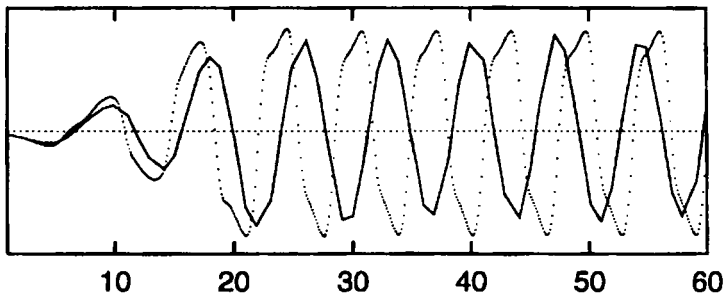

Figure 14. Second velocity component for channel flow at an interior point behind plate, calculated with CC-n ($k = 1{\cdot}0$, ————), compared with reference solution (·····)

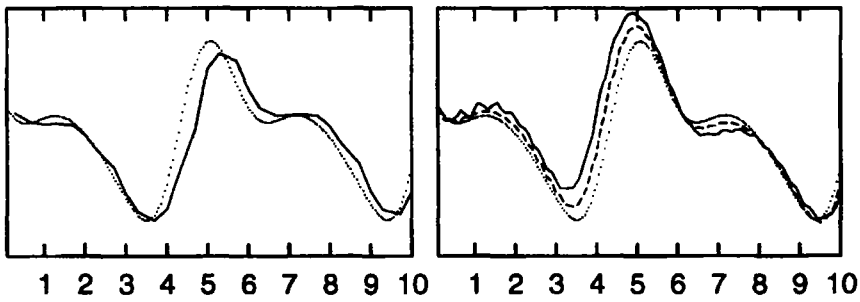

Figure 15. Mean pressure drop for channel flow, calculated by CC-n (left, $k = 0{\cdot}33$, ————) and PP-n (right) with $k = 0{\cdot}11$ (————) and 0·075 (– – – –), compared with reference solution (·····)

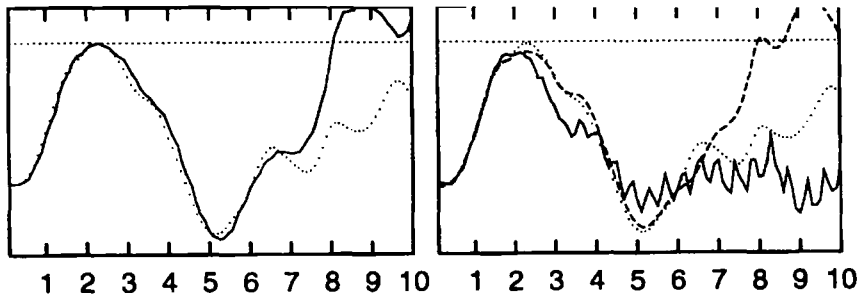

Figure 16. Total flux for Venturi pipe problem, calculated by CC-n (left, $k = 0{\cdot}11$, ————) and PP-n (right) with $k = 0{\cdot}05$ (————) and 0·022 (– – – –), compared with reference solution (·····)

Table I. Results for channel flow with $1/\nu = 5$ (c = coarse, f = fine)

Method	Time steps	Min. time step	Max. time step	CPU time (s)
CC-c (stationary)	**1**	—	—	**115**
CC-c	**72**	0·0176	3230·0	**2694**
CP-c	**87**	0·0118	525·0	**1540**
PP-c	**90**	0·0033	680·0	**1355**
CC-f (stationary)	**1**	—	—	**756**
CC-f	**81**	0·0139	3970·0	**13,480**
CP-f	**96**	0·0072	131·0	**8972**
PP-f	**111**	0·0014	876·0	**7883**

Table II. Results for channel flow with $1/\nu = 500$: averaged time step accepted by hand

Method	Start = Stokes, $T = 60$	Start = fully developed, $T = 10$
CC-c-500	**0·111**	**0·222**
PP-c-500	**0·044**	**0·055**
CC-f-500	**0·088**	**0·111**
PP-f-500	**0·033**	**0·033**

Table III. Results for channel flow with adaptive control: accepted time step by adaptive control unit $T = 10$ (start = fully developed)

Method	Min. time step	Max. time step	Ave. time step
CC-c-50	0·008	0·165	**0·119**
PP-c-50	0·004	0·090	**0·063**
CC-c-500	0·076	0·111	**0·085**
PP-c-500	0·032	0·054	**0·041**
CC-c-10,000	0·028	0·111	**0·041**
PP-c-10,000	0·024	0·053	**0·035**
CC-f-50	0·007	0·143	**0·109**
PP-f-50	0·003	0·074	**0·032**
CC-f-500	0·057	0·111	**0·070**
PP-f-500	0·020	0·042	**0·027**
CC-f-10,000	0·007	0·040	**0·024**
PP-f-10,000	0·004	0·028	**0·022**

Table IV. Results for Venturi pipe with $1/\nu = 1000$: averaged time step accepted by hand

Method	Start = Stokes, $T = 30$	Start = fully developed, $T = 10$
CC-c-1000	**0·044**	**0·050**
PP-c-1000	**0·022**	**0·022**
CC-f-1000	**0·033**	**0·045**
PP-f-1000	**0·010**	**0·011**

Table V. Results for Venturi pipe with adaptive control: accepted time step by adaptive control until $T = 10$ (start = fully developed)

Method	Min. time step	Max. time step	Ave. time step
CC-c-1000	0·016	0·065	**0·034**
PP-c-1000	0·012	0·040	**0·024**
CC-f-1000	0·005	0·033	**0·014**
PP-f-1000	0·033	0·019	**0·010**

gain in efficiency[11] for the linear algebra part is at least a factor of nine, we can claim that the projection schemes are faster than the coupled ones despite the fact that the total number of time steps may be larger. On the other hand, at the moment it is not clear to us which method is preferable: the pure projection method PP ($L = 1$) or the scheme CP which obtains the coupled solution via the splitting techniques ($L \geqslant 1$). The final discussion of the total efficiency will follow at the end of this section.

Another surprising observation is that all solutions obtained are 'identical' (up to round-off errors) even up to the boundary. This could mean that the projection methods do not exhibit spurious boundary oscillations of the pressure at all. However, our projection methods with $L = 1$ suffer from these defects, but significantly only for very small Reynolds numbers. For increasing Reynolds numbers these defects are invisible. One possible explanation might be that not only the width of these boundary layers is of order $O(kv)$, but also the absolute size of the error is proportional vk. This corresponds to first theoretical results and will be the subject of a forthcoming paper.

4.4. *Differences between fully non-linear and linearization techniques*

As explained in the previous section, we have different possibilities to treat the non-linear advective term: we can use fully implicit non-linear iteration schemes (version 'n'), we can perform a semi-implicit linearization using the previous solutions (versions 'c' and 'l') or we can treat this term fully explicitly by treating it as part of the right-hand side (using solutions from the previous time level, version 'x'). It is obvious that the fully implicit treatment leads to the most robust and accurate scheme, but the numerical work is the largest too. Analogous conclusions can be drawn for the other schemes. Hence again the question is: what are the required time steps in a practical example?

We perform the same test results as before (see Table VI and VII) and, for comparison only, we repeat the results for version 'n'. We drop the coupled versions CC-x, since no gain in efficiency could be reached (quasi-Stokes instead of Oseen problems, which lead to almost the same multigrid convergence rates), and the mixed schemes CP, which are presented later. In this test the resulting time steps for CP are about the same as for versions CC. We can see that the fully implicit coupled schemes CC-n lose their high accuracy as soon as we skip the 'exact' non-linear treatment: the linearized coupled schemes CC-l may need about the same time step size as the fully non-linear decoupled scheme PP-n. The differences for the projection schemes between fully non-linear PP-n and the linearized version PP-l are less pronounced. However, there is an difference in stability: owing to their more explicit character, the linearized versions tend to exhibit numerical oscillations when time steps are too large (see Figure 17).

Table VI. 'Linearization' results for channel flow with $1/\nu = 500$

Accepted time step by hand until $T = 60$ (start = Stokes)			
Method	Ave. time step	Method	Ave. time step
CC-n-c	**0·1111**	CC-n-f	**0·0800**
CC-l-c	**0·0750**	CC-l-f	**0·0650**
PP-n-c	**0·0444**	PP-n-f	**0·0333**
PP-l-c	**0·0333**	PP-l-f	**0·0222**
Accepted time step by hand until $T = 10$ (start = fully developed)			
Method	Ave. time step	Method	Ave. time step
CC-n-c	**0·2222**	CC-n-f	**0·1111**
CC-l-c	**0·0750**	CC-l-f	**0·0700**
PP-n-c	**0·0555**	PP-n-f	**0·0333**
PP-l-c	**0·0444**	PP-l-f	**0·0333**
PP-x-c	**0·0075**	PP-x-f	**0·0050**
Accepted time step by adaptive control until $T = 10$ (start = fully developed)			
Method	Min. time step	Max. time step	Ave. time step
CC-n-c	0·0762	0·1111	**0·0853**
CC-l-c	0·0333	0·0877	**0·0764**
PP-n-c	0·0321	0·0537	**0·0417**
PP-l-c	0·0329	0·0517	**0·0398**
PP-x-c	0·0048	0·0111	**0·0067**
CC-n-f	0·0574	0·1111	**0·0708**
CC-l-f	0·0301	0·0781	**0·0648**
PP-n-f	0·0196	0·0422	**0·0278**
PP-l-f	0·0199	0·0395	**0·0262**
PP-x-f	0·0035	0·0071	**0·0051**

Table VII. 'Linearization' results for Venturi pipe with $1/\nu = 1000$

Accepted time step by hand until $T = 30$ (start = Stokes)			
Method	Ave. time step	Method	Ave. time step
CC-n-c	**0·0500**	CC-n-f	**0·0450**
CC-l-c	**0·0333**	CC-l-f	**0·0333**
PP-n-c	**0·0222**	PP-n-f	**0·0150**
PP-l-c	**0·0111**	PP-l-f	**0·0090**
Accepted time step by hand until $T = 10$ (start = fully developed)			
Method	Ave. time step	Method	Ave. time step
CC-n-c	**0·0500**	CC-n-f	**0·0450**
CC-l-c	**0·0333**	CC-l-f	**0·0333**
PP-n-c	**0·0222**	PP-n-f	**0·0111**
PP-l-c	**0·0111**	PP-l-f	**0·0099**
PP-x-c	**0·0022**	PP-x-f	**0·0011**
Accepted time step by adaptive control until $T = 10$ (start = fully developed)			
Method	Min. time step	Max. time step	Ave. time step
CC-n-c	0·0158	0·0650	**0·0335**
CC-l-c	0·0059	0·0544	**0·0253**
PP-n-c	0·0122	0·0401	**0·0235**
PP-l-c	0·0135	0·0419	**0·0225**
PP-x-c	0·0001	0·0067	**0·0019**
CC-n-f	0·0054	0·0333	**0·0139**
CC-l-f	0·0029	0·0333	**0·0118**
PP-n-f	0·0035	0·0192	**0·0109**
PP-l-f	0·0057	0·0333	**0·0104**
PP-x-f	0·0001	0·0022	**0·0009**

The following efficiency results, which measure the total CPU time needed, might indicate that PP-l is to be preferred. However, calculations on very fine meshes, for the same or even higher Reynolds numbers, show different solutions obtained by PP-n and PP-l for the tolerance parameter $\varepsilon = 10^{-3}$. We believe that this is due to a lack of time step control and a smaller value for ε should be used. Then the necessary time steps k seem to be smaller for PP-l than for the non-linear version PP-n. However, this fact seems to be significant only in the case of high Reynolds numbers and very fine meshes, such that important non-linear effects can be really resolved. However, this very fine mesh width is far from being

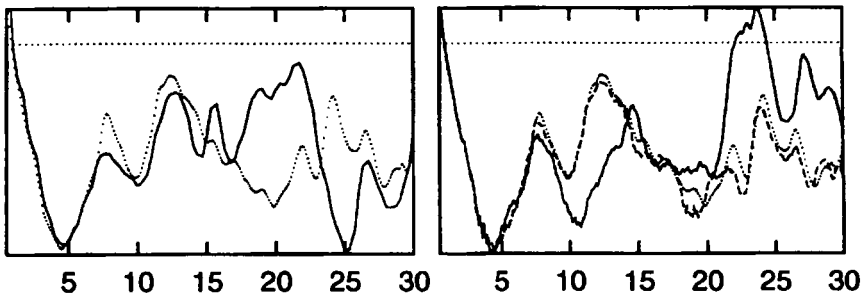

Figure 17. Pressure at an interior point for Venturi pipe, calculated by CC-n (left) with $k = 0{\cdot}33$ (————) and CC-l (right) with $k = 0{\cdot}05$ (————) and $0{\cdot}033$ (– – – –), compared with reference solution (·····)

realistic if we think of 3D applications on workstations, so the linearized approach seems to be favourable under the present conditions.

In contrast, the fully explicit treatment of the non-linear term always leads to much smaller step sizes. The choice has to be made for stability reasons only: for time steps slightly larger, numerical oscillations appear and the solutions blow up. We will see in the following considerations of the total numerical complexity that no gain in efficiency is reached (at least on our workstations), although the resulting linear systems are symmetric.

4.5. Differences in total efficiency

Finally we consider the total numerical efficiency, which is probably the most important measure for the quality of the methods proposed. We have already examined the necessary time step sizes for all schemes to guarantee approximately the same solution quality. Additionally we derived in Reference 11 theoretical results for the numerical complexity of each iteration step. The last open question is: evaluating all previous results, what total CPU time is needed to reach a given time level with a prescribed accuracy? To provide a fair comparison, we did all implementation by ourselves, so that the level of code optimization is about the same. Further, all codes are written for workstations only, neglecting vectorization and parallelization strategies.

Tables VIII and IX show the results until $T = 30$, starting from the Stokes solution as initial value and using an adaptive time step control. In comparison with the previous results, the versions 'CP' are performed too. In this case each non-linear iteration is stopped if one digit of accuracy is reached in the improvement of the defect (with $L \leqslant 9$), and only one multigrid sweep is performed. The maximum number of non-linear steps is $N = 9$. Therefore the versions 'CC' and 'CP' may differ, but the time step control guarantees (almost) the same results for velocity and pressure. In Figures 18 and 19 we present the graphs of resulting time steps for CC-n, PP-l and PP-x.

We finish this section with the following results. The fully implicit coupled schemes CC-n are the most accurate solvers, but also the most expensive ones for highly non-stationary flows. The methods which use splitting techniques reach a given time level much faster. They approximately satisfy the same prescribed accuracy despite the fact that the total number of time steps may be larger. On the other hand, at the moment it is not clear to us which method is preferable: the pure projection methods (PP, $L = 1$) or

Table VIII. Results for channel flow with $1/\nu = 500$

Method	Min. time step	Max. time step	Ave. time step	Elapsed time
CC-n-c	0·0296	0·2001	**0·0982**	**21,646**
CC-l-c	0·0186	0·1403	**0·0683**	**16,139**
CP-n-c	0·0289	0·1823	**0·0931**	**6464**
CP-l-c	0·0187	0·1182	**0·0669**	**7470**
PP-n-c	0·0139	0·1056	**0·0549**	**6511**
PP-l-c	0·0138	0·1021	**0·0539**	**4471**
PP-x-c	0·0001	0·0139	**0·0070**	**25,653**
CC-n-f	0·0262	0·1638	**0·0702**	**124,113**
CC-l-f	0·0171	0·1191	**0·0547**	**94,145**
CP-n-f	0·0246	0·1451	**0·0662**	**44,796**
CP-l-f	0·0157	0·0969	**0·0463**	**49,802**
PP-n-f	0·0099	0·0822	**0·0347**	**43,898**
PP-l-f	0·0099	0·0797	**0·0338**	**30,059**
PP-x-f	0·0001	0·0125	**0·0052**	**148,263**

Table IX. Results for Venturi pipe flow with $1/\nu = 1000$

Method	Min. time step	Max. time step	Ave. time step	Elapsed time
CC-n-c	0·0108	0·0594	**0·0271**	**167,202**
CC-l-c	0·0104	0·0407	**0·0183**	**139,724**
CP-n-c	0·0102	0·0502	**0·0267**	**32,669**
CP-l-c	0·0046	0·0391	**0·0223**	**62,913**
PP-n-c	0·0025	0·0468	**0·0195**	**33,896**
PP-l-c	0·0018	0·0404	**0·0190**	**25,306**
PP-x-c	0·0001	0·0049	**0·0017**	**221,392**
CC-n-f	0·0053	0·0384	**0·0138**	**1460,054**
CC-l-f	0·0050	0·0253	**0·0093**	**1852,784**
CP-n-f	0·0058	0·0333	**0·0137**	**307,691**
CP-l-f	0·0046	0·0317	**0·0083**	**1407,385**
PP-n-f	0·0014	0·0333	**0·0110**	**288,414**
PP-l-f	0·0011	0·0341	**0·0099**	**253,857**
PP-x-f	0·0001	0·0016	**0·0008**	**2362,262**

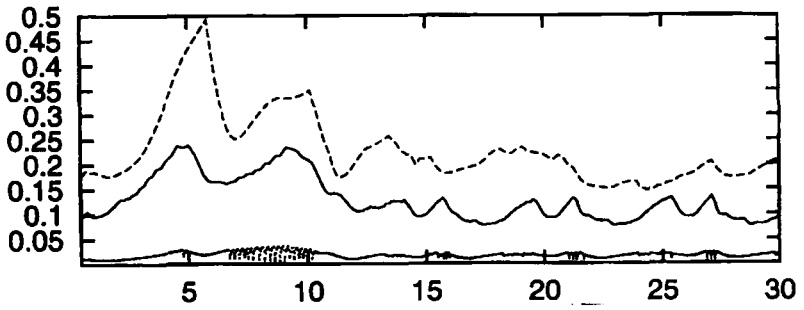

Figure 18. Adaptively chosen macro time step sizes $K = 3k$ for channel flow: methods CC-n (- - - -), PP-l (points) and PP-x (————)

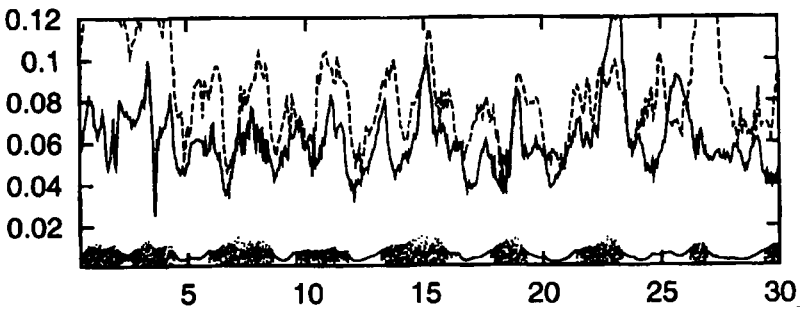

Figure 19. Adaptively chosen macro time step sizes $K = 3k$ for Venturi pipe: methods CC-n (- - - -), PP-l (————) and PP-x (points)

the schemes which obtain the coupled solution via the splitting techniques (CP, $L \geqslant 1$). Our present favourite is the linearized scheme PP-l.

5. CONCLUSIONS

Our intention in this paper is to compare (theoretically and numerically) some time-stepping schemes for the solution of the incompressible Navier–Stokes equations. They key points are the effect of some important components of the methods on their stability, accuracy and efficiency. These components are

(a) backward Euler, Crank–Nicolson or fractional-step θ scheme as discretization methods for the momentum equations
(b) coupled solvers in $\mathbf{u}$ and p or operator-splitting methods of projection type
(c) fully non-linear techniques, semi-implicit linearization or explicit treatment of the advection.

Theoretical considerations in Reference 11 show that the *discrete projection* approach in combination with *non-conforming finite elements* is well suited for these non-linear indefinite equations. The discrete projection schemes encompass $\mathbf{u}$–p coupled solution approaches as well as operator-splitting schemes. Particularly for highly non-stationary flows the numerical cost of the decoupled solvers is much lower and no robustness problems occur on anisotropic grids. However, the required time step size is smaller for these splitting schemes. Since an exact quantitative prediction seems to be impossible for 'real life' problems, we perform test calculations for a class of problems thought to be representative.

1. The flow in a channel around an inclined plate for Reynolds number (approximately $Re = 500$. An important physical quantity is the pressure distribution on the surface of the plate. This calculation corresponds to a typical mid-range-Reynolds-number problem and produces a periodically oscillating vortex shedding.
2. The flow in a Venturi pipe for Reynolds number (approximately) $Re = 5000$. Important physical quantities are the flux through the upper small inlet and the pressure distribution at the wall. The resulting solutions are very complex (in space and time) and no simple periodicity can be observed.

Our scheme represent for a large variety of methods: fully implicit non-linear coupled solvers; semi-implicit iteration methods of SIMPLE type; backward Euler, Crank–Nicolson and fractional-step θ schemes; projection schemes of first and second order (similar to the schemes of Chorin, Van Kan and Gresho); schemes with an explicit treatment of the advection often used in spectral codes; and some more. Therefore we hope that our comparison is somehow illustrative, especially in view of giving a general rating of all these schemes concerning accuracy, stability and efficiency. Additionally, with our special finite element discretization we can include other discretization schemes too. For instance, our favourite schemes of projection type lead to similar matrices as some classical staggered grid finite difference discretizations, but generalized for arbitrary grids. Furthermore, they include the complete finite element analysis by background.

We conclude with the recommendation to use splitting techniques in the fully non-stationary case of high Reynolds numbers, in which they may lead to much more robust and efficient methods than fully coupled approaches. In combination with the fractional-step θ scheme they give excellent results, with or without fully non-linear treatment. By theoretical considerations it seems to be preferable to use them as solver only (version CP, $L \geqslant 1$) to calculate the coupled solution, but our numerical tests indicate that the pure projection schemes (version PP, with $L = 1$) may be even more efficient. The fully coupled approach CC with coupled solution methods should be used only for low Reynolds numbers, in which case direct stationary solvers can be developed.

Summarizing, our recommendations for computing non-stationary flows are the following.

1. Use only second-order methods in time. Schemes such as backward Euler, the Chorin projection scheme and constant extrapolation in time for the advection should be avoided. The resulting time steps are much smaller and no essential gain in efficiency is achieved.
2. We suggest to use the fractional-step θ scheme. It has the same numerical complexity and accuracy as the Crank–Nicolson scheme but its stability behaviour is better.
3. Use the coupled solver only for low Reynolds numbers (i.e. for almost stationary solutions). For increasing Reynolds numbers the projection solvers are getting much better and these solvers are robust even on very anisotropic grids. For high Reynolds numbers the necessary time steps for the projection schemes are only moderately smaller (by a factor of two to five) compared with the coupled approach. Additionally, in this case no significant boundary layers of the pressure are visible and all obtained solutions are the same (up to round-off errors).
4. Do not treat the non-linear term fully explicitly. Despite the fact that the resulting linear systems are symmetric, the total CPU time is larger. Since very efficient solvers for non-symmetric schemes have recently become available and since the resulting time steps can be chosen larger (by at least a factor of 10), we prefer the fully non-linear or at least linearly extrapolated treatment. We propose the fully implicit methods with the adaptive defect correction approach because of their high robustness and accuracy. Further, the resulting linear subproblems can be solved very efficiently, since it is completely sufficient to gain one digit per iteration. Only in combination with very regular grids, i.e. tensor product meshes, might the explicit methods be preferable, at least on vector computers. However, this approach does not work on general domains and meshes. Furthermore, we are not sure about the relation between mesh size and time step for these methods. For the implicit schemes this coupling seems to be very weak and is controlled only by accuracy and not by robustness limits.
5. The tests performed show that an adaptive time step control is absolutely necessary. The reason is not only to obtain accurate solutions but also to exclude numerical instabilities. Only the fully implicit non-linear coupled approaches CC-n and CP-n in combination with the fractional-step θ scheme lead always to numerically 'stable' results. Without any time step control the projection-type solvers may be even worthless.

In Reference 11 we have discussed that full efficiency is achievable particularly with the non-conforming finite element spaces. This fact, together with the theoretical analysis which shows that these elements seem to be one of the simplest but most stable finite elements on arbitrary grids,[19] makes them attractive for practical flow simulation. First numerical tests in 3D with the coupled techniques[16] show their reliability in the general case. Our next step will be to add the developed splitting techniques and to perform similar 3D tests (see also Reference 1). Further, it would be very interesting to know whether our results can be reproduced with other spatial discretizations (conforming finite elements, finite volumes, etc.) and for other (probably more interesting) classes of problems, particularly for the 3D case. The author would be thankful for further suggestions and results.

REFERENCES

1. P. Schreiber, 'A new finite element solver for the non-stationary incompressible Navier–Stokes equations in three dimensions', *Thesis*, University of Heidelberg, 1996.
2. C. Johnson, 'The streamline diffusion finite element method for compressible and incompressible fluid flow', in *Finite Element Method in Fluids VII*, University of Alabama Press, Huntsville, AL, 1989.
3. O. Pironneau, 'On the transport–diffusion algorithm and its applications to the Navier–Stokes equations', *Numer. Math.*, **38**, 309–332 (1982).

4. R. Glowinski, 'Viscous flow simulation by finite element methods and related numerical techniques', in G. Murman and S. Abvarfavel (eds), *Progress and Supercomputing in Computational Fluid Dynamics*, Birkhauser, Boston, MA, 1985, pp. 173–210.
5. S. Turek, 'Tools for simulating non-stationary incompressible flow via discretely divergence-free finite element models', *Int. j. numer. methods fluids*, **18**, 71–105 (1994).
6. P. M. Gresho, 'On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix, Part 1: Theory, Part 2: Implementation', *Int. j. numer. methods fluids*, **11**, 587–620, 621–659 (1990).
7. S. P. Vanka, 'Implicit multigrid solutions of Navier–Stokes equations in primitive variables', *J. Comput. Phys.*, **65**, 138–158 (1985).
8. R. Rannacher, 'On Chorin's projection method for the incompressible Navier–Stokes equations', in R. Rautmann *et al.* (eds), *Navier–Stokes Equations: Theory and Numerical Methods*, Springer, Berlin, 1992.
9. V. Girault and P. A. Raviart, *Finite Element Methods for Navier–Stokes Equations*, Springer, Berlin, 1986.
10. R. Rannacher and S. Turek, 'A simple non-conforming quadrilateral Stokes element', *Numer. Methods Partial Diff. Eq.*, **8**, 97–111 (1992).
11. S. Turek, 'On discrete projection methods for the incompressible Navier–Stokes equations: an algorithm approach', *Tech. Rep. 70, SFB 359*, University of Heidelberg, 1994.
12. S. Müller, A. Prohl, R. Rannacher and S. Turek, 'Implicit time-discretization of the non-stationary incompressible Navier–Stokes equations', in G. Wittum and W. Hackbusch (eds), *Proc. 10th GAMM-Seminar*, Kiel, January 1994, Vieweg, Braunschweig, 199x.
13. M. O. Bristeau, R. Glowinski and J. Periaux, 'Numerical methods for the Navier–Stokes equations: applications to the simulation of compressible and incompressible viscous flows', *Rep. UH/MD-4*, University of Houston, 1987.
14. P. Kloucek and F. S. Rys, 'On the stability of the fractional-step θ scheme for the Navier–Stokes equations', *SIAM J. Numer. Anal.*, **31**, 1312–1335 (1994).
15. M. Crouzeix and P. A. Raviart, 'Conforming and non-conforming finite element methods for solving the stationary Stokes equations', *RAIRO*, **R-3**, 77–104 (1973).
16. P. Schreiber and S. Turek, 'An efficient finite element solver for the non-stationary incompressible Navier–Stokes equations in two and three dimensions', *Proc. Workshop on Numerical Methods for the Navier–Stokes Equations*, Heidelberg, October 1993, Vieweg, Braunschweig, 199x.
17. S. Turek, 'Modern finite element solution techniques for the incompressible Navier–Stokes equations. A practical guide with computational details', in preparation.
18. G. Wittum, 'The use of fast solvers in computational fluid dynamics', in P. Wesseling (ed.), *Numerical Methods in Fluid Mechanics*, Vieweg, Braunschweig, 1990.
19. R. Becker and R. Rannacher, 'Finite element discretization of the Stokes and Navier–Stokes equations on anisotropic grids', in G. Wittum and W. Hackbusch (eds), *Proc. 10th GAMM-Seminar*, Kiel, January 1994, Vieweg, Braunschweig, 199x.
20. M. S. Engleman, V. Haroutunian and I. Hasbani, 'Segregated finite element algorithms for the numerical solution of large-scale incompressible flow problems', *Int. j. numer. methods fluids*, **17**, 323–348 (1993).
21. A. J. Chorin, 'Numerical solution of the Navier–Stokes equations', *Math. Comput.*, **22**, 745–762 (1968).
22. J. Van Kan, 'A second-order accurate pressure-correction scheme for viscous incompressible flow', *SIAM J. Sci. Stat. Comput.*, **7**, 870–891 (1986).
23. C. Johnson, R. Rannacher and M. Boman, 'Numerics and hydrodynamic stability: towards error control in CFD, *Preprint 1993–13*, Mathematics Department, Chalmers University of Technology, 1993.
24. J. Heywood, R. Rannacher and S. Turek, 'Artificial boundaries and flux and pressure conditions for the incompressible Navier–Stokes equations', *Int. j. numer. methods fluids*, **22**, 325–352 (1996).
25. R. Rannacher, 'Numerical analysis of the Navier–Stokes equations', *Appl. Math.*, **38**, 361–380 (1993).