Fraunhofer Institut
Techno- und
Wirtschaftsmathematik

O. Iliev, D. Stoyanov

Multigrid – adaptive local refinement solver for incompressible flows

Berichte des Fraunhofer ITWM, Nr. 54 (2003)

ISSN 1434-9973

Bericht 54 (2003)

Die Veröffentlichungen in der Berichtsreihe des Fraunhofer ITWM können bezogen werden über:

Fraunhofer-Institut für Techno- und
Wirtschaftsmathematik ITWM
Gottlieb-Daimler-Straße, Geb. 49

67663 Kaiserslautern

Telefon: +49 (0) 6 31/2 05-32 42
Telefax: +49 (0) 6 31/2 05-41 39
E-Mail: info@itwm.fraunhofer.de
Internet: www.itwm.fraunhofer.de

Vorwort

Das Tätigkeitsfeld des Fraunhofer Instituts für Techno- und Wirtschaftsmathematik ITWM umfasst anwendungsnahe Grundlagenforschung, angewandte Forschung sowie Beratung und kundenspezifische Lösungen auf allen Gebieten, die für Techno- und Wirtschaftsmathematik bedeutsam sind.

In der Reihe »Berichte des Fraunhofer ITWM« soll die Arbeit des Instituts kontinuierlich einer interessierten Öffentlichkeit in Industrie, Wirtschaft und Wissenschaft vorgestellt werden. Durch die enge Verzahnung mit dem Fachbereich Mathematik der Universität Kaiserslautern sowie durch zahlreiche Kooperationen mit internationalen Institutionen und Hochschulen in den Bereichen Ausbildung und Forschung ist ein großes Potenzial für Forschungsberichte vorhanden. In die Berichtreihe sollen sowohl hervorragende Diplom- und Projektarbeiten und Dissertationen als auch Forschungsberichte der Institutsmitarbeiter und Institutsgäste zu aktuellen Fragen der Techno- und Wirtschaftsmathematik aufgenommen werden.

Darüberhinaus bietet die Reihe ein Forum für die Berichterstattung über die zahlreichen Kooperationsprojekte des Instituts mit Partnern aus Industrie und Wirtschaft.

Berichterstattung heißt hier Dokumentation darüber, wie aktuelle Ergebnisse aus mathematischer Forschungs- und Entwicklungsarbeit in industrielle Anwendungen und Softwareprodukte transferiert werden, und wie umgekehrt Probleme der Praxis neue interessante mathematische Fragestellungen generieren.

Prof. Dr. Dieter Prätzel-Wolters
Institutsleiter

Kaiserslautern, im Juni 2001

Multigrid – adaptive local refinement solver for incompressible flows

Oleg Iliev, Dimitar Stoyanov *

Abstract

A non-linear multigrid solver for incompressible Navier-Stokes equations, exploiting finite volume discretization of the equations, is extended by adaptive local refinement. The multigrid is the outer iterative cycle, while the SIMPLE algorithm is used as a smoothing procedure. Error indicators are used to define the refinement subdomain. A special implementation approach is used, which allows to perform unstructured local refinement in conjunction with the finite volume discretization. The multigrid - adaptive local refinement algorithm is tested on 2D Poisson equation and further is applied to a lid-driven flow in a cavity (2D and 3D case), comparing the results with bench-mark data. The software design principles of the solver are also discussed.

Keywords: Navier-Stokes equations, incompressible flow, projection-type splitting, SIMPLE, multigrid methods, adaptive local refinement, lid-driven flow in a cavity

1 Introduction

The numerical solution of nonlinear equations, such as Navier-Stokes equations, requires significant computational efforts, even on modern computers. Therefore, special attention is paid to the development of efficient numerical algorithms for solving such problems. The multigrid method and local refinement techniques are among the most powerful tools for accelerating flow computations. Multigrid algorithms for linear and nonlinear problems are actively developed during the last decades (see, for example, [6, 15, 3] and references therein). We use the full approximation storage (FAS) approach which extends the linear multigrid to non-linear problems. In the case of stationary equations FAS is usually combined with the so called full multigrid (FMG) scheme, in which the solution computed on the current grid is used as an initial guess on the next finer grid. An essential feature of the MG method is that it is an optimal iterative method - the number of iterations does not depend on the number of unknowns. The local refinement (LR) technique makes the numerical algorithms

*Fraunhofer Institut fuer Techno- und Wirtschaftsmathematik, Gottlieb-Daimler-Str., Geb.49, D-67663 Kaiserslautern, Germany, e-mail: {iliev, stoyanov}@itwm.fhg.de

for PDEs more efficient: accurate results are computed using less resources. LR technique is often combined with MG method (or other multilevel approach) aiming at preserving the optimal character of the method, and at the same time, at reducing the CPU and the memory usage. The essential questions for LR technique are where exactly to refine the grid, and how to discretize and efficiently to solve the problem on the composite (locally refined+remaining coarse) grid. The first question concerns the related a posteriori error estimators and/or error indicators. A good review in this field is given in [14], see also the references therein. In this paper we just describe the indicators we use and concentrate on answering the second question. MG-LR techniques were used, among others, by McCormick [9], Ewing et al [2], etc. While the local refinement approach is well developed for linear elliptic and parabolic problems, its application to systems of nonlinear equations is still under active development. An interesting question is how to organize the interaction of the decoupling (splitting) of the system and of the nonlinearity iterations with the local refinement strategy. Here we consider an algorithm for dealing with this in the case when a projection method, namely SIMPLE, is used for decoupling the system of incompressible Navier-Stokes equations, and the treatment of the decoupling on the interfaces between the coarse and the refined parts of the domain needs a special attention. More precisely, the paper describes the local refinement features of an incompressible flow solver, based on the full multigrid–full approximation storage algorithm. Finite volume method (method of balance) [12] is used for discretization of the velocity – pressure formulation of 2D/3D Navier-Stokes equations on cell-centered grids. SIMPLE algorithm is used as a smoother within the global MG algorithm.

The solver described here has been designed using an object oriented hierarchy. The exploited structuring and type of hierarchy provide certain advantages in the case of the adaptive local refinement algorithm. In particular, because one of the basic data types corresponds to a control volume (CV), it is possible to refine even within a single CV, as well as to have different levels of refinement in different subdomains.

The paper is organized as follows. The next section describes the numerical algorithms employed in the solver: the SIMPLE method for Navier-Stokes equations and the discretizations schemes used, then the MG techniques, further treating their local refinement extension in more details. Then we discuss some basic principles for the software design of PDE solvers in general. Results from numerical experiments are presented: first the MG-solution of a real physical (non-isothermal) problem are discussed, and then the applications of the MG-LR algorithm are demonstrated - after a validation of the solver for 2D Poisson equation, it has been applied to an incompressible flow in a 2D and 3D cavity. Finally some conclusions are drawn.

2 A SIMPLE based MG–LR algorithm

2.1 Governing equations

Consider the steady state incompressible Navier-Stokes equations:

$$\frac{\partial}{\partial x_j}(\rho u_j) = 0 \tag{1}$$

$$\frac{\partial}{\partial x_j}(\rho u_j u_i - \mu \frac{\partial u_i}{\partial x_j}) = -\frac{\partial p}{\partial x_i} + f_i, \qquad i,j = 1,2,3. \tag{2}$$

Here $\mathbf{u} = (u_1, u_2, u_3)^t = (u, v, w)^t$ stands for the velocity vector, x_j are Cartesian coordinates, ρ stands for density, p is the pressure, μ is the viscosity and f_i are the body forces. Appropriate boundary conditions complete the system. The summation convention over repeating indices is exploited above.

2.2 SIMPLE as a Navier-Stokes single-grid solver

Consider a computational domain, which is a connected union of control volumes (CVs), where each CV is a brick (for brevity we consider 2D case). All the unknowns are related to the control volume centers, i.e. the so called collocated arrangement of the unknowns is used. The equations are discretized in a finite volume manner, and are linearized in a fixed point manner, i.e. the velocities in the convective terms are taken from the preceding iteration (see more details further, see also [4, 3]). Thus, at each step of an iterative process one has to solve

$$\begin{pmatrix} Q_1(\mathbf{u}) & 0 & \partial^{\mathrm{h}}_{x_1} \\ 0 & Q_2(\mathbf{u}) & \partial^{\mathrm{h}}_{x_2} \\ \partial^{\mathrm{h}}_{x_1} & \partial^{\mathrm{h}}_{x_2} & 0 \end{pmatrix} \begin{pmatrix} u_1^{k+1} \\ u_2^{k+1} \\ p^{k+1}. \end{pmatrix} = \begin{pmatrix} F_1 \\ F_2 \\ 0 \end{pmatrix}. \tag{3}$$

where $k+1$ stands for the number of the iteration to be performed. Q_1, Q_2 denote linear operators, including both convective and diffusive terms of the discretized momentum equations. Upwind or central differencing can be used for the convective terms. Notation $\partial^{\mathrm{h}}_{x_i} p^k$ stands for the $i-$th component of the discrete gradient operator. The SIMPLE algorithm (Semi-Implicit Method for Pressure Linked Equations) has been proposed by Patankar and Spalding [10] and it is widely used presently for solving incompressible viscous flows. We briefly describe this splitting algorithm following the formalism of Fletcher [4]. Each SIMPLE-iteration contains the following steps:

(1) Compute $\bar{u}^{k+1}, \bar{v}^{k+1}$ from $Q_1 \bar{u} = F_1 - \partial^{\mathrm{h}}_x p^k$, $Q_2 \bar{v} = F_2 - \partial^{\mathrm{h}}_y p^k$;
(2) Solve a Poisson-type equation for the pressure correction p';
(3) Calculate velocity corrections $\mathbf{u}' = -D^{-1} \nabla^{\mathrm{h}} p'$;
(4) Correct the velocities $\mathbf{u}^{k+1} = \bar{\mathbf{u}}^{k+1} + \alpha_u \mathbf{u}'$ and the pressure $p^{k+1} = p^* + \alpha_p p'$.

At the first step, the momentum equations are solved assuming the pressure is known from the preceding iteration. Thus one obtains an initial approximation $\bar{\mathbf{u}}$ for the velocity vector. (This initial approximation is often designated as $\vec{\mathbf{u}}^* = (u^*, v^*)^t$, we will also use such designation in the forthcoming more detailed description.) At the next step, the pressure correction equation (PCE) is formed aiming at satisfying the continuity equation. More precisely, one defines some velocity corrections through

$\vec{\mathbf{u}}' = -D^{-1}\nabla^{\mathrm{h}} p'$, and obtains PCE taking discrete divergence from $\mathbf{u}^{k+1} = \bar{\mathbf{u}}^{k+1} + \mathbf{u}'$, i.e. $\mathbf{u}^{k+1} = \bar{\mathbf{u}}^{k+1} - D^{-1}\nabla^{\mathrm{h}} p'$, where $D = diag\ (diagQ_1(u), diagQ_2(u))$. The obtained PCE looks as follows (see for details [4, 3])

$$\nabla^{\mathrm{h}} \mathbf{u}^{k+1} = -\nabla^{\mathrm{h}} D^{-1} \nabla^{\mathrm{h}} p' + \nabla^{\mathrm{h}} \bar{\mathbf{u}}^{k+1} = 0. \tag{4}$$

The expression $\nabla^{\mathrm{h}} \bar{\mathbf{u}}^{\mathbf{k+1}}$ is the so called mass source term. In the case of collocated grids special interpolation schemes [11] for the cell face velocities must be used in order to avoid non-physical oscillations. Underrelaxation parameters α_u, α_p are used to improve the convergence of the algorithm (see [10, 3] for details).

2.3 Discretization schemes

2.3.1 Finite volumes method

A representative control volume (CV) in an orthogonal grid is drawn on Fig. 2.3.1.1. - its central node is the point P and the CV index is (i, j, k). We assume, that the neighbour CVs in a certain space direction have neighbour indices, although this is not obligatory for our consideration.

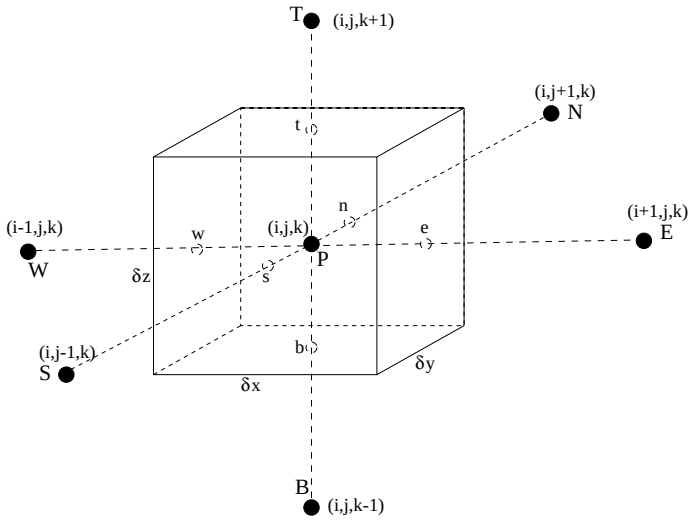

Fig. 2.3.1.1. A representative CV

The capital letters W, E, S, N, B, T designate the centers of the neighbour CVs in WEST, EAST, SOUTH, NORTH, BOTTOM and TOP directions correspondingly. The same (but lower case) letters mark the centers of the CV walls in each direction. The size of the edges of our representative CV are δx, δy, and δz: through them one defines the volume δV of the CV, and also the area of the CV faces in each direction $(A_w, A_e, A_s, A_n, A_b, A_t)$. One also assumes, that the volume integrals are calculated as a multiplication of the CV nodal value by the volume δV, while the surface integrals are equal to the product of the value in the wall center and the wall area. Thus for the momentum equations (2), after applying the Gauss divergence theorem, one obtains

the so called "general transport equation" (see [10]), valid for all transport variables ϕ (i.e. not for the velocity components only):

$$\begin{aligned}
&\frac{\partial(\rho\phi)}{\partial t}\delta V + \\
&\left\{\left[\rho_e u_e (\delta y\ \delta z)_e \phi_e - \Gamma_{\phi,e}\left(\frac{\partial\phi}{\partial x}\right)_e (\delta y\ \delta z)_e\right] - \left[\rho_w u_w (\delta y\ \delta z)_w \phi_w - \Gamma_{\phi,w}\left(\frac{\partial\phi}{\partial x}\right)_w (\delta y\ \delta z)_w\right]\right\} + \\
&\left\{\left[\rho_n v_n (\delta x\ \delta z)_n \phi_n - \Gamma_{\phi,n}\left(\frac{\partial\phi}{\partial y}\right)_n (\delta x\ \delta z)_n\right] - \left[\rho_s v_s (\delta x\ \delta z)_s \phi_s - \Gamma_{\phi,s}\left(\frac{\partial\phi}{\partial y}\right)_s (\delta x\ \delta z)_s\right]\right\} + \\
&\left\{\left[\rho_t w_t (\delta x\ \delta y)_t \phi_t - \Gamma_{\phi,t}\left(\frac{\partial\phi}{\partial z}\right)_t (\delta x\ \delta y)_t\right] - \left[\rho_b w_b (\delta x\ \delta y)_b \phi_b - \Gamma_{\phi,b}\left(\frac{\partial\phi}{\partial z}\right)_b (\delta x\ \delta y)_b\right]\right\} \\
&= S_\phi \delta V.
\end{aligned} \tag{5}$$

In the case the general transport variable ϕ is some of the velocity components u_i, $(i = 1, 2, 3)$, then $\Gamma_\phi = \mu$ (the dynamical viscosity) and $S_\phi = -\frac{\partial p}{\partial x_i} + f_i$. The equation (5) shows the way the system of equation (1),(2) is split: the pressure field approximation is considered to be known when the momentum equations are calculated. The other part of the splitted system, the pressure correction equation, will be discussed in the forthcoming paragraphs. The linearization of the momentum equations is of a fixed - point type: the convective terms are written in an equivalent non-divergent form, e.g. $\rho u_j \frac{\partial \phi_i}{\partial x_j}$, and the velocity components are considered to be known. The solution of the momentum equations can be of Jacobi-type - i.e. in each momentum equation one uses velocities approximations, obtained from the previous SIMPLE iteration, or of Gauss-Seidel-type, when the velocity components already computed, are used within the scope of the same iteration when the next momentum equations are to be linearized.

2.3.2 Discretization of the momentum equations

The momentum equation after an implicit discretization will have the following general form:

$$a_P\phi_P + \sum a_{nghb}\phi_{nghb} = a_P\phi_P + a_W\phi_W + a_E\phi_E + a_S\phi_S + a_N\phi_N + a_B\phi_B + a_T\phi_T = S_P. \tag{6}$$

The variables ϕ are the unknowns (ϕ_P, ϕ_{nghb}) at the CV nodal point P and at the nodes of the neighbours CVs (see Fig. 2.3.1.1.) with the corresponding coefficients a_P, a_{nghb}. S_P is the point source/sink term for the variable ϕ, located at the CV node. The system of linear equations obtained is solved most often by some iterative solver. The size of the system is equal to the total number of unknowns, i.e. the number of CV nodes - because one writes such an equation for each CV. Further, the values at each internal wall are obtained through a linear interpolation of the neighbour node values. We will give the discretization of both convective and diffusive terms in (2) for the east wall in the case of a regular orthogonal grid.

Discretization of the convective terms

We consider two types of discretizations: a second-order central difference scheme (CDS), and first order upwind differencing scheme (UDS). Other types of discretization, e.g. higher order upwind schemes, are also possible but are not discussed here. The convective term associated with the east wall is $(\rho\ u_e\ A_e)\phi_e$. If **the wall is internal**, i.e. the neighbour CV exists, the convective term produces contributions to a_P and a_E:

- **CDS:** the wall value is linearly interpolated through the nodal values:

$$\phi_e = (1-f)\phi_P + f\phi_E, \quad f = \frac{x_e - x_P}{x_E - x_P}. \tag{7}$$

  Therefore: $a_P \longleftarrow (1-f)(\rho\ u_e\ A_e)$ and $a_E \longleftarrow f(\rho\ u_e\ A_e)$.

- **UDS:** the contribution depends on the sign of u_e:

$$\phi_e = \begin{cases} \phi_P & \text{if } u_e > 0 \quad \Rightarrow a_P \longleftarrow (\rho\ u_e\ A_e) \text{ and } a_E \longleftarrow 0 \\ \phi_{nghb} & \text{if } u_e < 0 \quad \Rightarrow a_P \longleftarrow 0 \text{ and } a_E \longleftarrow (\rho\ u_e\ A_e) \end{cases}$$

In the case of a **Dirichlet boundary wall** the value ϕ_e is known from the boundary condition and the contribution is to the free term S_P only: $S_P \longleftarrow (-\rho\ u_e\ A_e)\phi_e$.

Discretization of the diffusive terms

Always second order central difference scheme (CDS) is used for the derivatives of the type $\left(\frac{\partial\phi}{\partial x}\right)_e$. In case of an internal wall this gives

$$\left(\frac{\partial\phi}{\partial x}\right)_e \approx \frac{\phi_E - \phi_P}{x_E - x_P},$$

producing contributions to a_P and to a_E coefficients. If the east CV wall is a boundary one, again central CDS is used, but having first order of approximation:

$$\left(\frac{\partial\phi}{\partial x}\right)_e \approx \frac{\phi_e - \phi_P}{x_e - x_P}.$$

Then again one has a contribution to a_P and to S_P for Dirichlet type of BCs. For Neumann/Robin and periodic types of BCs the contribution is to the coefficient, associated with the additional unknown in the boundary wall center, instead to S_P directly.

The diffusive term associated with the east wall is $-\Gamma_{\phi,e}\left(\frac{\partial\phi}{\partial x}\right)_e A_e$. Thus for an **internal wall** we get:

$$a_P \longleftarrow \Gamma_{\phi,e}\left(\frac{1}{x_E - x_P}\right)A_e \text{ and } a_E \longleftarrow -\Gamma_{\phi,e}\left(\frac{1}{x_E - x_P}\right)A_e.$$

And for a **Dirichlet boundary wall**, when ϕ_e is known from the boundary condition:

$$a_P \longleftarrow \Gamma_{\phi,e}\left(\frac{1}{x_e - x_P}\right)A_e \text{ and } S_P \longleftarrow \Gamma_{\phi,e}\left(\frac{\phi_e}{x_e - x_P}\right)A_e.$$

2.3.3 Relaxation schemes for the discretized momentum equations

Underrelaxation is often applied to stabilize the numerical solution of non–linear problems. We follow [10] to introduce it. One starts with the discretized variant (6) of the general transport equation, where the quantities, related to the neighbour nodes, have a subscript $_{nghb}$:

$$a_P \phi_P = -\sum a_{nghb} \phi_{nghb} + S_P.$$

This way one obtains

$$\phi_P = \frac{-\sum a_{nghb} \phi_{nghb} + S_P}{a_P}.$$

Adding and subtracting ϕ_P^* (i.e. the value of ϕ_P on the preceding iteration) in the right–hand side of the equation above, one gets:

$$\phi_P = \phi_P^* + (\frac{-\sum a_{nghb} \phi_{nghb} + S_P}{a_P} - \phi_P^*).$$

The expression in the brackets of the right–hand side of the latter equation contains the change of the value of ϕ_p , obtained in the current iteration. This change might be corrected by a coefficient α, i.e.

$$\phi_P = \phi_P^* + \alpha(\frac{-\sum a_{nghb} \phi_{nghb} + S_P}{a_P} - \phi_P^*). \tag{8}$$

The equation (8) - with a "relaxed" current step change - is the one, which is to be solved instead of the "standard" discretized equation (6). We have "over–relaxation" for $\alpha > 1$ and "under–relaxation" otherwise. Expanding the expression in (8), we obtain the changes, introduced in the original discretization coefficients (described so far), because of the relaxation. They are in the diagonal coefficient a_P and in the source term S_P only:

$$a_P \longleftarrow \frac{a_P}{\alpha}, \quad S_P \longleftarrow S_P + (1-\alpha)\frac{a_P}{\alpha}\phi_P^* \ .$$

The stabilizing effect of the underrelaxation is similar to the one achieved in the case of backward Euler discretization for the time derivative with a time step small enough: both methods provide a diagonal dominance of the matrix elements. Thus the relaxation approach in many cases betters significantly the stability of the numerical solution.

2.3.4 Pressure correction equation (PCE)

The approximation for the velocity, obtained after the solution of the momentum equations, does not satisfy the continuity equation. Therefore, using the continuity equation, one searches corrections for the velocities, in order the latter to fullfil the mass ballance requirement. The corrected velocity can be written as

$$\vec{\mathbf{u}} = \vec{\mathbf{u}}^* + \vec{\mathbf{u}}', \tag{9}$$

where $\vec{\mathbf{u}}^*$ is the solution of the momentum equations on the current step and $\vec{\mathbf{u}}'$ is the vector of the velocity corrections, which must be found. Before to derive the PCE we will find an useful relation between the pressure correction and the velocity corrections. This relation, together with (9), will be the starting point to derive PCE.

Relation between the velocity correction and the pressure correction

The discretized variant (4) of the general transport equation, written for a particular velocity component, say $u_1 = u$, gives:

$$a_P u_P = -\sum a_{nghb} u_{nghb} + S_P = -\sum a_{nghb} u_{nghb} + \Big(-\frac{p_e - p_w}{x_e - x_w} + f_1(x,y,z)\Big)\delta x \delta y \delta z.$$

The latter equation is satisfied by the vector $\vec{\mathbf{u}}^*$, which has been obtained as a solution of the equation above for $p_e = p_{e,old}$ and $p_w = p_{w,old}$ (the values for the pressure from the previous SIMPLE iteration). Therefore we have:

$$a_P u_P^* = -\sum a_{nghb} u_{nghb}^* + \Big(-\frac{p_{e,old} - p_{w,old}}{x_e - x_w} + f_1(x,y,z)\Big)\delta x \delta y \delta z. \qquad (10)$$

Now we want the corrected velocities from (9) and the corrected pressure $p = p_{old} + p'$ to satisfy the same generalized transport equation (6), with the same coefficients a. Thus we can write

$$a_P(u_P^* + u_P^{'}) = -\sum a_{nghb}(u_{nghb}^* + u_{nghb}^{'}) + \Big(-\frac{p_e - p_w}{x_e - x_w} + f_1(x,y,z)\Big)\delta x \delta y \delta z. \qquad (11)$$

If we now subtract eqn. (10) from eqn. (11), we get

$$a_P u_P^{'} = -\sum a_{nghb} u_{nghb}^{'} - \frac{p_e^{'} - p_w^{'}}{x_e - x_w}\delta x \delta y \delta z. \qquad (12)$$

The latter equation is the one we are searching for: it gives the velocity corrections at the CV nodes, providing the wall pressure corrections are known. On the other side this equation is obviously not efficient numerically: one obtains a full matrix of coefficients (a_P and a_{nghb}) before the unknowns, because the corrections $u_{nghb}^{'}$ at the nodes of the neighbour CVs are also not known. And here comes **the main approximation of the SIMPLE–algorithm**, namely the assumption that

$$\sum a_{nghb} u_{nghb}^{'} = 0.$$

This assumption is generally not correct. It had been initially introduced mainly because of the acceptable results obtained through it. Later on it has been mathematically grounded also, see more details in [13]. Through this simplification one gets the required relation

$$u_P^{'} = -\frac{1}{a_P}(p_e^{'} - p_w^{'})\delta y \delta z, \qquad (13)$$

used in the derivation of PCE. The coefficient a_P here comes from the discretization of the $u-$momentum equation in this CV. To avoid eventual misunderstanding, the momentum equation variable, this coefficient is related to, will be designated as a superscript, e.g. a_P^u.

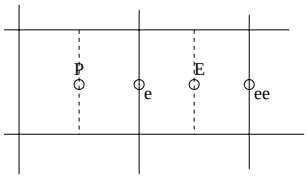

Fig. 2.3.4.1. 2D projection of the representative CV and its east neighbour

Few additional remarks follow:

(i) Different modifications of the SIMPLE–algorithm are obtained as a result of different approximations used for the term $\sum a_{nghb} u'_{nghb}$ in (12). For more details see [3];

(ii) We will use (13) in a bit different way: after the solution of PCE we obtain the pressure corrections at the CV nodes, but not at the walls, as (13) requires. From these nodal pressure corrections we must further obtain the velocity corrections: first on the walls and then at the CV nodes. That is why we need a formula which connects the wall velocity corrections with the nodal pressure correction. To find it - of course - the same idea may be used as in the derivation of (13). But the new relation might be directly written, if we consider (see Fig. 2.3.4.1.) an imaginary CV, having a node at the point e (the center of the EAST wall of our "representative" CV). The east and west walls of this imaginary CV then have as their centers the points P and E and are marked with a dashed line on Fig. 2.3.4.1. If we apply the same consideration for this imaginary CV, we obtain the same relation, but written in a way we need it:

$$u'_e = -\frac{1}{\bar{a}^u_{P,e}}(p'_E - p'_P)\delta y \delta z, \quad \bar{a}^u_{P,e} = (1-f)a^u_P + f a^u_{P,E}, \quad f = \frac{x_e - x_P}{x_E - x_P} \ . \qquad (14)$$

In the latter formula the coefficient $\bar{a}^u_P$ of the imaginary CV is linearly approximated by (7) using the values a_P of the "representative" CVs with nodes P and E;

(iii) In the same way one obtains the corresponding relations for the other walls of the representative CV.

The formula (14) shows that the wall velocity corrections depend on the nodal pressure corrections as well as on the discretization coefficients a_P (NB! this is the coefficient before the relaxation) in the adjacent CV nodes, obtained from the discretization of the momentum equations for the corresponding velocity component.

Derivation of the PCE

PCE is a discrete equation, which has no continuous prototype. In some sense the continuity eqn (1) could be considered as such a prototype, but due to the approxi-

mations, used to obtain PCE from it, this is far from a true conclusion. The PCE is derived from the continuity eqn. (1), the expression for the velocity corrections (9), and the relations (14) between the velocity corrections and the pressure correction. One starts by integrating the continuity eqn. (1) within the representative CV (its volume is $\delta V = \delta x \delta y \delta z$):

$$\int\int\int_{CV} \frac{\partial u}{\partial x} dxdydz + \int\int\int_{CV} \frac{\partial v}{\partial y} dxdydz + \int\int\int_{CV} \frac{\partial w}{\partial z} dxdydz = 0.$$

Applying the Gauss divergence theorem to the integrals in the right-hand side of the equation above, one obtains

$$\Big(u_e \delta y \delta z - u_w \delta y \delta z\Big) + \Big(v_n \delta x \delta z - v_s \delta x \delta z\Big) + \Big(w_t \delta x \delta y - w_b \delta x \delta y\Big) = 0.$$

The latter relation is valid both for the continuous variables and for the grid-function of the numerical solution for the velocities. Therefore, one can express the velocity components there through their initial approximations and their further corrections, according to (9). We get

$$\begin{aligned} &\Big(u_e^{'} \delta y \delta z - u_w^{'} \delta y \delta z\Big) + \Big(v_n^{'} \delta x \delta z - v_s^{'} \delta x \delta z\Big) + \Big(w_t^{'} \delta x \delta y - w_b^{'} \delta x \delta y\Big) = \\ &-\Big\{\Big(u_e^{*} \delta y \delta z - u_w^{*} \delta y \delta z\Big) + \Big(v_n^{*} \delta x \delta z - v_s^{*} \delta x \delta z\Big) + \Big(w_t^{*} \delta x \delta y - w_b^{*} \delta x \delta y\Big)\Big\}. \end{aligned} \tag{15}$$

The expression closed in the contour brackets in the right–hand side of (15) is the **mass source term**: this is the mass disballance within the CV for the velocity approximation, that we have obtained after the solution of the momentum equations. On the other side, the continuity equation (1) requires this mass disballance to be zero. Thus, we look for such velocity corrections, which make this mass disballance to vanish. We will further consider in details the calculation of the mass source term. In the following equation we just denote it by MST. The equation itself is obtained from (15) by replacing in its left-hand side the velocity corrections by the expressions relating them to the pressure corrections, like (14). Thus we obtain:

$$\begin{aligned} &\Big[-\tfrac{1}{\bar{a}_{P,e}^{u}}(p_E^{'} - p_P^{'})(\delta y)^2(\delta z)^2 + \tfrac{1}{\bar{a}_{P,w}^{u}}(p_P^{'} - p_W^{'})(\delta y)^2(\delta z)^2\Big] + \\ &\Big[-\tfrac{1}{\bar{a}_{P,n}^{v}}(p_N^{'} - p_P^{'})(\delta x)^2(\delta z)^2 + \tfrac{1}{\bar{a}_{P,s}^{v}}(p_P^{'} - p_S^{'})(\delta x)^2(\delta z)^2\Big] + \\ &\Big[-\tfrac{1}{\bar{a}_{P,t}^{w}}(p_T^{'} - p_P^{'})(\delta x)^2(\delta y)^2 + \tfrac{1}{\bar{a}_{P,b}^{w}}(p_P^{'} - p_B^{'})(\delta x)^2(\delta y)^2\Big] = -MST. \end{aligned}$$

This equation gives the discretization coefficients for the PCE. They are:

$$
\begin{aligned}
a_E &= -\tfrac{1}{\bar{a}^u_{P,e}}(\delta y)^2(\delta z)^2,\ a_W = -\tfrac{1}{\bar{a}^u_{P,w}}(\delta y)^2(\delta z)^2,\\
a_N &= -\tfrac{1}{\bar{a}^v_{P,n}}(\delta x)^2(\delta z)^2,\ a_S = -\tfrac{1}{\bar{a}^v_{P,s}}(\delta x)^2(\delta z)^2,\\
a_T &= -\tfrac{1}{\bar{a}^w_{P,t}}(\delta x)^2(\delta y)^2,\ a_B = -\tfrac{1}{\bar{a}^w_{P,b}}(\delta x)^2(\delta y)^2,\\
a_P &= -(a_E + a_W + a_N + a_S + a_T + a_B).
\end{aligned}
\tag{16}
$$

All averaged coefficients in (16), coming from the discretization of the momentum equations, are calculated through linear interpolation similar to the one in (14). One sees from (16) that PCE is a Poisson type equation.

We need also to add to the PCE coefficients (16) the free term, which is the mass source term taken with a negative sign, see (15). But the calculation of the mass source term needs an additional discussion, because of the following reason:
As we have mentioned earlier, the values at the walls are linearly interpolated through the nodal values (see (7)). So, the wall values for the velocities, used in (15) to calculate the mass source term, are also linearly interpolated. Further in (16) we linearly interpolate the values of the a_P coefficients from the momentum equations. If the pressure and the velocity calculations are performed on the same grid, this kind of interpolation creates non-physical oscillations (see e.g. [10]). To overcome the problem two ways are known: (i) to "shake" the grids for the pressure and the velocity calculations, i.e. to use staggered grids ([10]); or (ii) to use a special kind of interpolation for the wall velocities ([11]) in the mass source term, still using collocated grids for both the pressure and the velocity computations. We present here the latter approach. In order to find such cell-face velocity approximations, which - after putting them in the right - hand side of (15) - do not cause non-physical oscillations, the requested interpolation should take into account not only the velocity values in the neighbouring nodes, but the pressure also.

Cell–face velocities

We will derive the expression for the cell-face velocity u_e on the east wall (it is "starred" in (15), here we skip the star). We consider the representative CV (with a point P as a center) and its east neighbour (center at the point E), pictured on Fig. 2.3.4.1. We use our usual notation, only adding the corresponding superscripts ^{P} or ^{E} to designate the CV the corresponding quantity belongs to. Let us start the derivation from the $u-$momentum equations in these CVs, but writing separately in the expressions for the source terms S^P_P and S^E_P those parts, which contain the pressure:

$$
u_P = u^P_P = \frac{1}{a^P_P}\Big[-\sum a^P_{nghb} u^P_{nghb} + F^P_1 - (p^P_e - p^P_w)\delta y \delta z \Big], \tag{17}
$$

$$
u_E = u^E_P = \frac{1}{a^E_P}\Big[-\sum a^E_{nghb} u^E_{nghb} + F^E_1 - (p^E_e - p^E_w)\delta y \delta z \Big]. \tag{18}
$$

We also know that u_e is a linear interpolation (see (7)) from u^P_P and u^E_P, i.e.:

$$
u_e = (1-f)u^P_P + f u^E_P, \quad f = \tfrac{x_e - x_P}{x_E - x_P}. \tag{19}
$$

By inserting (17) and (18) in (19), one obtains:

$$u_e = (1-f)\left\{ \tfrac{1}{a_P^P}\left[-\sum a_{nghb}^P u_{nghb}^P + F_1^P \right] - \tfrac{\delta y \delta z}{a_P^P}(p_e^P - p_w^P) \right\} + f\left\{ \tfrac{1}{a_P^E}\left[-\sum a_{nghb}^E u_{nghb}^E + F_1^E \right] - \tfrac{\delta y \delta z}{a_P^E}(p_e^E - p_w^E) \right\}.$$

Let us write the last equation as

$$\begin{aligned} u_e = (1-f)&\left\{ \tfrac{1}{a_P^P}\left[-\sum a_{nghb}^P u_{nghb}^P + F_1^P \right] \right\} \\ &+ f\left\{ \tfrac{1}{a_P^E}\left[-\sum a_{nghb}^E u_{nghb}^E + F_1^E \right] \right\} \\ &- \delta y \delta z \left[(1-f)\tfrac{1}{a_P^P}(p_e^P - p_w^P) + f\tfrac{1}{a_P^E}(p_e^E - p_w^E) \right]. \end{aligned} \tag{20}$$

Note, that now the last expression in the right–hand side of (20) is the linear interpolation averaging (taken at the point e) of the quotient of the pressure difference and the a_P coefficient, i.e:

$$(1-f)\frac{1}{a_P^P}(p_e^P - p_w^P) + f\frac{1}{a_P^E}(p_e^E - p_w^E) = \overline{\left(\frac{p_e - p_w}{a_P}\right)}_e.$$

Taking into account that $\overline{(p_e - p_w)_e} = p_E - p_P$, one can write the last term in (20) as:

$$\begin{aligned} -\delta y \delta z &\left[(1-f)\tfrac{1}{a_P^P}(p_e^P - p_w^P) + f\tfrac{1}{a_P^E}(p_e^E - p_w^E) \right] = \\ &-\delta y \delta z (p_E - p_P)\overline{\left(\tfrac{1}{a_P}\right)}_e = -\delta y \delta z (p_E - p_P)\left[(1-f)\tfrac{1}{a_P^P} + f\tfrac{1}{a_P^E} \right]. \end{aligned} \tag{21}$$

In a similar way we transform the first two terms in the right–hand side of (20). We have:

$$\begin{aligned} &\tfrac{1}{a_P^P}\left[-\sum a_{nghb}^P u_{nghb}^P + F_1^P \right] = \\ &\tfrac{1}{a_P^P}\Big[\underbrace{-\sum a_{nghb}^P u_{nghb}^P + F_1^P - (p_e^P - p_w^P)\delta y \delta z - a_P^P u_P}_{=0,\text{ from (17)}} + (p_e^P - p_w^P)\delta y \delta z + a_P^P u_P \Big]. \end{aligned}$$

Or, for the first term in the right–hand side of (20), we finally get:

$$\frac{1}{a_P^P}\left[-\sum a_{nghb}^P u_{nghb}^P + F_1^P \right] = \left[u_P^P + \frac{\delta y \delta z}{a_P^P}(p_e^P - p_w^P) \right] = \left[u_P + \frac{\delta y \delta z}{a_P^P}(p_e^P - p_w^P) \right]. \tag{22}$$

For the second term there, following the same sequence of derivations, we write

$$\frac{1}{a_P^E}\left[-\sum a_{nghb}^E u_{nghb}^E + F_1^E\right] = \left[u_P^E + \frac{\delta y \delta z}{a_P^E}(p_e^E - p_w^E)\right] = \left[u_E + \frac{\delta y \delta z}{a_P^E}(p_{ee} - p_e)\right]. \quad (23)$$

For the designation p_{ee} in the last equality see Fig. 2.3.4.1. (this is the east wall of the east neighbour). Now, substituting (21), (22) and (23) in (20), we get the required approximation for the east cell-face velocity, i.e.

$$\begin{aligned} u_e = (1-f)\left[u_P + \tfrac{\delta y \delta z}{a_P^{u,P}}(p_e^P - p_w^P)\right] + \left[u_E + \tfrac{\delta y \delta z}{a_P^{u,E}}(p_e^E - p_w^E)\right] \\ -\delta y \delta z (p_E - p_P)\left[(1-f)\tfrac{1}{a_P^{u,P}} + f\tfrac{1}{a_P^{u,E}}\right], \qquad f = \tfrac{x_e - x_P}{x_E - x_P}. \end{aligned} \quad (24)$$

In (24) we have added to the coefficient a_P another upperscript ^{u} in order to designate the discretization of which momentum equation is referred to. Following the same style of derivation, one obtains similar approximations for each of the cell-face velocities.

2.3.5 Relaxation of the pressure updates

No relaxation is used to solve PCE. But it is used later: when, after the solution of PCE, one must obtain the values for the pressure. Then, instead of $p = p_{old} + p'$, one uses $p = p_{old} + \alpha_p p'$, where α_p is the (under-)relaxation coefficient for the pressure. This approach also betters the overall convergency of the algorithm. Note, that this pressure relaxation is to be done **after** the velocity corrections have been already calculated, i.e. just before starting the new SIMPLE iteration. Thus, the sequence of the steps, associated with the PCE, could be summarized as follows:

(i) solve PCE and obtain pressure corrections at the CV nodes. Then the wall pressure corrections might be found as linear interpolations through the nodal corrections;

(ii) find the velocity corrections (at the nodes and on the walls), using (13) and (14);

(iii) correct the velocities and the pressure.

2.4 Multigrid algorithm

The basic idea of the multigrid approach is that the high-frequency amplitudes of the error are reduced (or "smoothed") on the fine grid(s), while the low-frequency error amplitudes are reduced on the coarse grid(s). Thus, the multigrid methods combine the positive effects on both fine and coarse grids and the main result is a fast convergence. Here we consider briefly the FMG–FAS approach, which will be further used as a basis for the local refinement algorithm. Consider the problem $Lu = f$. L here is a discrete operator, u and f are grid functions. When one solves single equation (e.g. Poisson in our case) u is a scalar, and the smoother is some iterative method with smoothing properties, such as Jacoby, Gauss-Seidel, ILU, etc. When

Navier-Stokes equations are to be solved, one should consider u being a vector, i.e. $u \equiv (\mathbf{u}, p)^t$. In this latter case L is a nonlinear operator, and SIMPLE–algorithm is used as a smoother. In both cases (when u is a scalar or a vector) the problem has been discretized over a set of grids G^k, $k = 0, ..N$, where k is the grid index and for the coarsest grid $k = 0$. A short description of the non-linear multigrid algorithm, written in a pseudo-code manner of [15], follows:

(1) procedure NonLinearMGalgorithm(k, $\overset{o}{u}^{\mathrm{k}}, u^{\mathrm{k}}, AST^{\mathrm{k}}$)
/* k: current grid index; $\overset{o}{u}^{\mathrm{k}}$: initial guess for solution (vector); */
/* u^{k}: output solution (vector); AST^{k}: Additional source term (vector);*/

(2) if(k=0) then /* coarsest grid */
(3) Smooth(k, $\overset{o}{u}^{\mathrm{k}}, u^{\mathrm{k}}, 4(\nu_1 + \nu_2), AST^{\mathrm{k}}$)
(4) else begin /* finer grid, $k > 0$ */
(5) Smooth(k, $\overset{o}{u}^{\mathrm{k}}, \tilde{u}^{\mathrm{k}}, \nu_1, AST^{\mathrm{k}}$)
(6) $r^{\mathrm{k}} = f^{\mathrm{k}} + AST^{\mathrm{k}} - L^{\mathrm{k}}(\tilde{u}^{\mathrm{k}})$
(7) $\tilde{u}^{\text{k-1}} = R\tilde{u}^{\mathrm{k}}$; $\tilde{r}^{\text{k-1}} = Rr^{\mathrm{k}}$
(8) $AST^{\text{k-1}} = L^{\text{k-1}}(\tilde{u}^{\text{k-1}}) - f^{\text{k-1}} + \tilde{r}^{\text{k-1}}$
(9) NonLinearMGalgorithm($k-1, \tilde{u}^{\text{k-1}}, u^{\text{k-1}}, AST^{\text{k-1}}$)
(10) $c^{\text{k-1}} = u^{\text{k-1}} - \bar{u}^{\text{k-1}}$
(11) $c^{\mathrm{k}} = Pc^{\text{k-1}}$; $\bar{u}^{\mathrm{k}} = \tilde{u}^{\mathrm{k}} + c^{\mathrm{k}}$
(12) Smooth(k, $\bar{u}^{\mathrm{k}}, u^{\mathrm{k}}, \nu_2, AST^{\mathrm{k}}$)
(13) end

Let us briefly comment the algorithm when the Navier-Stokes eqns. are solved. On the coarsest grid, a modification of the above described single grid SIMPLE algorithms works (line (3)). The modification concerns accounting for an additional non-zero source term vector (denoted AST), coming from the full approximation scheme. Certain number (ν_1) of smoothings (called pre-smoothings) are carried out on each finer grid (line (5)). Line (6) calculates the current finer grid residual, here $AST^{\mathrm{k}} = 0$ if k is current finest grid, otherwise this term comes as a parameter, calculated on the finer grid. The restriction of the current residuals and current solution approximation on the next coarser grid follows on line (7). The restriction operator for the solution is a weighted average operator over all fine grid control volumes which form the current coarse grid control volume. The restriction operator for the residuals is just a sum over these finer grid CVs residuals. This kind of restriction follows directly from the conservation laws. The additional source term vector on the coarser grid is calculated on line (8). On line (10) the coarser grid correction vector is calculated. On line (11) the prolongation of the coarser grid corrections to the finer grid is done using bi- or three-linear interpolation, and then the finer grid solution approximation is obtained as a sum of the pre-smoothed solution plus prolongated coarser grid correction. At last, some number, (ν_2), of post-smoothing iterations are performed.

The FMG approach exploits the coarser grid steady solution as a good initial guess on the next finer grid. Successively, the solution computed on the current grid is prolongated and it is used as an initial guess on the next finer grid. On each grid MG-sweeps are performed until the prescribed stop criterion is satisfied.

2.5 Adaptive local refinement algorithm as an extension of FMG-FAS

In general, a composite grid (fine in the refined subdomain plus coarse in the remaining part) has to be considered in conjunction with the local refinement. Approaches for LR could be conditionally split into two groups with respect to the chosen way for discretization on the composite grid. These approaches are sometimes equivalent at an abstract level, but their implementation is quite different. In the first approach, the governing equations are discretized explicitly on the composite grid, further multigrid (multilevel) methods and/or domain decomposition technique can be used for efficient solution of the obtained system of algebraic equations (see, for example, [2]). In the second approach, first a domain decomposition is performed at a continuous level, and after that the governing equations are discretized in each subdomain separately (see, for example, [9]). In the latter case, no explicit discretization on the composite grid is done, therefore the accuracy and the efficiency of the algorithm depend on the prescribed interface conditions. In fact, the second approach can be viewed as a variant of the Schwartz algorithm combined with usage of different types of grids in different subdomains. Here we will discuss the second from the above mentioned LR approaches, which from our point of view is more suitable for applying the MG-LR strategy.

2.5.1 A MG–LR algorithm for Poisson equation

A LR scheme based on FMG algorithm for Poisson equation is discussed in [9] for the case of vertex based grid, it is known as "bordered multilevel scheme". A modification of this approach to cell-centered grids can be found in [2]. An explicit discretization on a composite grid is presented there, combined with algebraic multilevel iterative procedure for solving the resulting linear algebraic system. Here we present a different approach. We add an auxiliary layer of coarse grid control volumes, laying outside the refined subdomain. Note, the fine grid–smoother works only within the refined subdomain, but on the lower grid levels the so called additional source term in FAS has to be calculated for the auxiliary CVs as well. Due to the chosen approach with the auxiliary layers, the same discretization, as in the standard MG, is used everywhere. Moreover, there is no need to change anything in the discretization, when non-neighbouring levels of refinement are used in neighbouring subdomains.

2.5.2 MG-LR algorithm and SIMPLE

In this case the numerical algorithm involves a decomposition of the domain according to the chosen local refinement approach, and splitting of the Navier–Stokes equations

in SIMPLE algorithm. Thus, the question is: Should we first decompose and after that split, or it is better first to split, and after that to decompose. In other way, the question can be formulated as follows: is it better first to organize LR for the system and after that to apply SIMPLE in each subdomain, or it is better first to apply SIMPLE to the system of Navier-Stokes equations, and after that to use LR approach for each equation separately. The both approaches differ in prescribing the conditions on the interface between the refined and the non-refined subdomains. We investigated two different types of interface conditions, corresponding to the both cases mentioned above. In the first case, after the current computations on the coarse grid are completed and the velocity is prolongated, a local problem within the refined subdomain is treated, prescribing the velocity on the interface. This Dirichlet type condition for the velocity implies zero Neumann boundary conditions for the pressure correction equation on the interface. In the second case, we consider LR approach for velocity and PCE equation separately, i.e. supposing the splitting of the system has been already done. However, in this case the boundary conditions are prescribed not directly on the interface, but in the centers of the auxiliary CVs. In this case, Dirichlet b.c. for the velocity in auxiliary nodes is used when solving momentum equations, and zero Dirichlet b.c. in auxiliary nodes is used when solving PCE.

In our numerical experiments we got no stable convergence of the MG-LR algorithm when the first variant was applied. All the results presented further are computed exploiting the second variant.

2.5.3 Error indicators

The appropriate choice of a posteriori local estimators is a strongly problem-dependent task, often with no obvious solution. We use a rather simple approach, as in [1]. Two types of sensors are discussed there. The first one is a criterion based on the solution gradient or a related property:

$$Cr = |\nabla\phi|_{min} + \gamma(|\nabla\phi|_{max} - |\nabla\phi|_{min}), \tag{25}$$

where $\nabla\phi$ is the gradient of the coarse grid solution and γ is a free parameter. The second one is an estimate of the solution error, based on a Richardson extrapolation:

$$e^h(x_1, x_2) = \frac{\phi^h(x_1, x_2) - \phi^{2h}(x_1, x_2)}{2^p - 1}. \tag{26}$$

In the latter equation h is the step size and p is the order of the discretization method. The Richardson error, calculated in (26) is then used in (25) instead of the gradient $\nabla\phi$. Thus certain CV is to be refined on the next grid level, if the locally calculated indicator exceeds the prescribed value of Cr for some γ given.

3 Software design of the solver

Among the software technologies the Object Oriented Programming (OOP) - as a highest level of data and program organization - offers more possibilities for a robustness and elegance in the programming of CFD algorithms than the trivial and

rather old structuring approaches. The basic idea of OOP is that each object type (class) encapsulates its own data and the corresponding procedures processing the latter. The objects can inherit each other and the whole program is build up as a hierarchy of objects. However, the OOP-technology itself does not lead automatically to an efficient software design: it is just a tool. Therefore, an essential question in the OOP-design is how to define the objects and to build-up the hierarchy of the classes in the program. Here we propose a hierarchy skeleton of classes (objects) for the development of a general-type numerical solvers for systems of PDEs. It is intended for solvers which are based on "local type" numerical methods and discretizations as Finite Difference Method (FDM), Finite Volume Method (FVM), Finite Elements Method (FEM). The methods can be both implicit or explicit. "Global methods" (classical Galerkin, spectral methods etc.), leading to dense matrices, are excluded from our consideration. The concrete implementation of the class hierarchy of the solver follows these principles.

3.1 Leading principles

Local discretization: All "local" methods could be formulated in terms of local discretization and further assembling of these local approximations (local matrices) to produce a global matrix. This is a FEM terminology, but it is a suitable generalization for our purposes. In these methods we are searching for local discretizations within some **Basic Discretization Element** (BDE). In each particular case this is the grid node (FDM), the control volume (FVM), or the finite element (FEM). Thus a basic generic object in our hierarchy of classes should correspond to this generalized item - the Basic Discretization Element. Its concrete interpretations in the case of a particular numerical method will lead to a concrete concept with a concrete characteristics (e.g. discretization approaches).

Grid topology, fictive and dummy Basic Discretization Elements: We suppose the grid is given in advance, e.g. obtained through a grid generator. In some problems the grid can change (or to be modified) on each time step. Within the grid some "fictive" basic discretization elements might exist: they contain a "prescribed" solution (i.e. known in advance, or such BDEs have known discretization coefficients) - as in the fictitious domain methods and similar.

Multilevel features: The general pattern for single-grid solution procedures might be further extended for a set of (nested) grids in the case of multilevel algorithms, where on each next level the grid is successively refined (locally or globally). The single-grid solvers, working on a certain level, are rather independent elements of these multilevel algorithms. Another important characteristic is that some inter-grid exchange always exists between the grid levels.

Therefore, we could state the following leading principles of the structure design:

1. Local discretization with further assembling of a "global" matrix (at the subdomain or domain level). Thus the type (class) Basic Discretization Element must

be considered as basic one from the point of view of the numerical algorithm;

2. The grid topology must be handled only through implicit (i.e. non-direct) links between the neighbour Basic Discr Elements. This will allow more flexibility in handling the grid, and eventually its subdivision into blocks during the parallel processing, further reshaping of these blocks, etc;

3. A natural requirement is the designed hierarchy of objects to follow the hierarchy of the concepts of the numerical algorithms.

3.2 The hierarchy of classes

For each grid we construct a global structure - the **grid topology information table** (`gridInfoTable`). In it the BDEs for the current grid are successively indexed. For each of these BDEs an entry exists in the Grid Topology Information Table (GTIT), corresponding to BDE's index, and a pointer (`pBDE`) is kept to the memory, associated with this BDE. The information contained in the table is used to resolve all "topological" questions, i.e. to search neighbours of certain BDE. This can be done e.g. if each entry in the GTIT-table (corresponding to certain BDE) keeps pointers to the GTIT-entries of neighbour BDEs. Therefore, the neighbour links of BDEs are not direct: they are carried through the tables GTIT.

Further, we design the program hierarchy of objects in a way that there is one-to-one correspondence with the hierarchy of concepts of the numerical algorithm. For the solver presented here the basic classes (levels) and the encapsulated proper procedures therein are:

- **CV-level, Basic Discretization Element** (`TBscDiscrEl`): the domain description, problem/equation description, BC settings, discretization of the governing equations are performed here;

- **GRID-level** (`TGrid`): the SIMPLE solver/smoother works here;

- **INTER-GRID-level:** the FMG-FAS algorithm (i.e. the intergrid communication control) is associated with it.

Basic Discretization Element class characteristics: First, all the information about the boundary and interface conditions is stored locally within the BDE description. Second, the local approximation for the differential operator is also locally stored - it is obtained through the methods, encapsulated within the class `TBscDiscrEl`, a set of procedures performing the discretizations. The most important principle for performing the local discretization is that each derivative (or term in the equation) should be described separately. The discretization stencil for the differential operator is then assembled step by step, i.e. derivative by derivative or term by term, instead of being programmed at once for the whole operator (as it often happens). Thus one can use a library of elementary stencils and the whole operator is described as references to such elementary stencils. This makes more flexible the description of the differential operator we are trying to represent in a discrete form, i.e. the model description itself becomes easier within the code.

The leading principle is that all information which directly concerns the local approximations of the differential operator within certain BDE is stored in the data

and procedures of the basic type `TBscDiscrEl`. When one performs a loop over all BDEs within a block (or a grid), one obtains the local discretization matrices.

Grid class characteristics (`TGrid`): The loop over all BDEs is performed on the next level: within the class `TGrid`. The class instance "knows" the set of BDEs belonging to the grid through the list of integers, which refer to the indices of the grid topology info table (GTIT) at a certain grid level.

The procedure for solving a single equation within a system of coupled PDEs (we presuppose a splitting, as in SIMPLE) is often the same for all variables. Some slight differences are possible due to the model or the numerical method, but they are rather not essential from implementational point of view. The numerical algorithm for the solution of the whole system (in our case SIMPLE) consists of successive references to the procedure `solveSingleEqn` but with different arguments for the different variables. The procedure itself has few clearly defined main steps:

```
int TGrid::solveSingleEqn(VAR, DISCR_STENCIL)
// VAR: the parameter, indicating the variable we are solving for
// DISCR_STENCIL: parameter indicating which discr. stencil to be used
{
   int i, globMatrSz;

   globMatrSz=0;  // the size of the global matrix (number of unknown grid-nodes)
   for(i=0; i < MAX_BDE; i++) { // a loop over all BDEs in this grid

          // if this is a dummy BDE, see next i
        if(gridInfoTable[i].pBDE == NULL) continue;

                      // otherwise search for a local discretization
        gridInfoTable[i].pBDE->calcDiscrCoeffs(VAR, DISCR_STENCIL);

        globMatrSz++;           // increase the global matrix size counter
   }

   assembleGlobalMatrix();      // assebling the global matrix
   solveLinearSystem();         // solving the linear system

   return 1;
};
```

The procedure has as parameters `VAR` - the variable we are solving for; and `DISCR_STENCIL` - a key word, which defines how to discretize (some terms in) the equation, e.g. in our case central differences or upwind for the convective terms in (2). The pointers `gridInfoTable[i].pBDE` in the GTIT refer to the BDEs within the grid, i.e. to instances of the class `TBscDiscrEl`, which contains the encapsulated procedure `TBscDiscrEl::calcDiscrCoeffs(VAR, DISCR_STENCIL)` performing the local discretization. In a loop over all BDEs within the grid one obtains the local approximations, then the global matrix is assembled and further the system of (linear) algebraic equations is solved.

Note, that only a slight modification is needed in order to handle explicite methods (without assebling). This modification concerns not even the structure, but only

the method `TBlock::solveSingleEqn(..)`. There, after obtaining the local matrices (explicitly), one should directly (locally) calculate the solution on the next time step, instead of assembling a global matrix and further solving a system of algebraic equations.

Inter-Grid class characteristics: The highest level in the object hierarchy handles the inter-grid communications. The inter-grid transfer operators are defined here, etc. The single-grid solvers, which are elements of the multilevel strategy, are handled on the lower `TGrid` level and here they are referred to. Actually on this level the FMG-FAS algorithm is coded in a way which follows almost directly the pseudo-code description from the previous sections.

3.3 Advantages of this design

1. A possibility is provided for setting up some rules for the model description - term by term, using the desired stencil as a parameter and chosen from a library of stencils. This makes easy eventual modifications of the model;

2. The principle to store locally - within each BDE - the information about the geometry, boundary and interface conditions, etc., makes the corresponding part of the model description also local. Therefore the settings, related to this part of the mathematical model, follow some clearly defined rules, which allow their easier modification also.

3. The rule for local discretization and global assembling makes methods like finite differences or finite volumes to obtain similar flexibility as the one available in the finite elements methods.

4. The parts of the numerical algorithm are situated on different levels. These different levels are clearly distinguished from a logical, but also from **an implementational** point of view, i.e. the object orientation provides (as a tool!) the necessary logical discipline in the process of implementation. Therefore, this clear distinction is not just an elegant logical construction, but it rather provides a lot of real advantages. The main advantage of such a design is that some more essential changes would affect the solver only on a particular level. Moreover, the new procedures can be simply added as new "methods" to the existing ones on the corresponding level, preserving the general structure of the whole code.

5. Possibilities to extend this organization for parallel machines.

4 Numerical experiments

The solver has been extensively tested on different 2D/3D flow problems in complex geometries. Now we present results related to the multigrid solution of a real physical problem and results from the verification of the adaptive refinement algorithm.

4.1 Simulation of a floor heating in a room

We consider a non-isothermal flow within a "furnished" room. The aim is to catch just the main features of the process, therefore the model is not that detailed: turbulent effects are not considered. In this case to the governing equations in the system (1), (2) we should add the heat transfer equation

$$\frac{\partial}{\partial x_j}(\rho u_j T - \frac{\mu}{Pr}\frac{\partial T}{\partial x_j}) = 0, \tag{27}$$

where T is the temperature and Pr is the Prandtl number. The body forces in the momentum equations (2) use Boussinesq approximation to model the natural convection, i.e.

$$f_i = -\rho_0 g_i \beta (T - T_0), \qquad i = 1, 2, 3. \tag{28}$$

The coefficients g_i in (28) account for the earth acceleration projection over the coordinate axes. In our case $g_1 = g_2 = 0$ and $g_3 = g_z = g = 9.81[m/s^2]$. The reference density ρ_0 is related to the reference temperature T_0 and β is a thermal expansion coefficient.

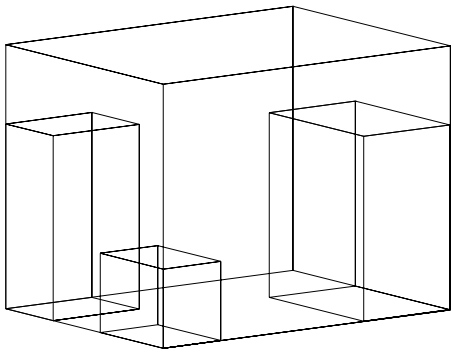

Fig. 4.1.1. Room geometry

The geometry of the room (Fig. 4.1.1.) is a parallelepiped of 4x3x3 [m] which contains furniture - some parallelepiped subdomains with no flow inside - as follows: (i) (0-0.8, 0-1.2, 0-0.9) : first furniture subdomain in x-, y- and z-direction; (ii) (0-1.2, 2.1-3, 0-2.1) : second furniture subdomain; (iii) (2.8-4, 0-1.8, 0-2.1) : third subdomain. On all of the walls (including the furniture) non-slip boundary conditions are applied for the velocity components. The temperature on all walls (including all furniture walls) satisfy Neumann zero boundary conditions with exception of non-furnished east wall and non-furnished bottom wall. On these walls Dirichlet type boundary condition is applied: $T_{bottom} = T_{hot}$ and $T_{east} = T_{cold}$.
The thermo-physical constants have been chosen in a way, which gives a Grashof number $Gr = 2.88e + 5$. For the single grid solver under-relaxation coefficients (might be not the optimal ones) $\alpha_p = \alpha_u = \alpha_v = \alpha_w = 0.75$, and $\alpha_T = 0.5$ have been chosen for the SIMPLE algorithm. The same coefficients have been also used for the multigrid

algorithm.

Table 4.1.1. Single grid (SG) and FMG-FAS solution history, ε=1e-4

	[40,40,40] SG	[10,10,10]	[20,20,20]	[40,40,40]	[80,80,80]
start rsd	5.60000e-04	2.2400e-03	7.0236e-04	3.5010e-04	1.0501e-4
end rsd	5.59761e-08	2.2122e-07	6.9430e-08	3.1589e-08	9.8358e-9
SIMPLE itr.	2865	399	31	23	22
MG-sweeps	-	-	3	2	2

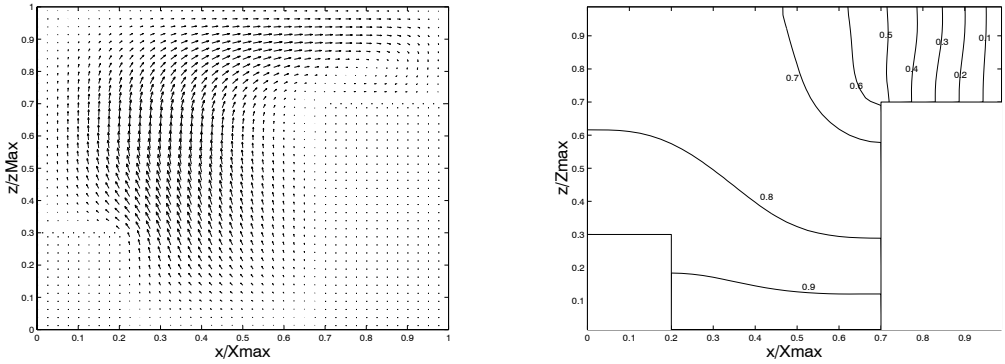

Fig. 4.1.2. Velocity field (left) and temperature isolines in the cross-section $y/Ymax = 0.25$

The multigrid discretization uses four grids, finest one of 80^3 CVs. The residuals are calculated as a vector sum of the C-norms of the velocity and the temperature. They are reduced 10^4 times on each grid. The comparison between the single grid and the multigrid convergence history could be seen on Table 4.1.1. The multigrid acceleration is impressive compared to the single-grid convergence rate. Fig. 4.1.2. represents the velocity field and the dimensionless temperature isolines in the cross-section plane $y/Ymax = 0.25$. The large "white" areas are the furniture obstacles.

4.2 Adaptive local refinement, 2D problems

The adaptive MG-LR algorithm is validated in solving 2D Poisson equation, as well as in computing lid-driven cavity flow for $Re = 100$ and $Re = 400$.

4.2.1 Validation of the MG–LR algorithm for 2D Poisson equation

The boundary conditions and the right hand side of the discretized equation correspond to an exact solution in the form:

$$u(x, y, a, b, \xi, \eta) = e^{-\frac{(x-\xi)^2}{a^2} - \frac{(y-\eta)^2}{b^2}}.$$

Table 4.2.1.1. 2D Poisson equation, single peak in (0,0)

	Local refinement			Multigrid		
grid level	3	4	5	3	4	5
CVs	*1400*	*5040*	*19816*	*6400*	*25600*	*102400*
MG-sweeps	6	6	10	6	6	6
$\|\text{appr-exact}\|_C$	1.74e-3	4.36e-4	1.09e-4	1.8e-3	4.6e-4	1.1e-4

Table 4.2.1.2. 2D Poisson equation, peaks in (0,0) and (0.5,1)

	Local refinement			Multigrid		
grid level	2	3	4	2	3	4
CVs	*1588*	*4700*	*15184*	*1600*	*6400*	*25600*
MG-sweeps	6	6	10	6	6	6
$\|\text{appr-exact}\|_C$	7.14e-3	1.77e-3	4.44e-4	7.14e-3	1.77e-3	4.44e-4

Table 4.2.1.1. presents results when $a = b = 0.15$, $\xi = \eta = 0$, i.e. the solution has a single peak in (0,0). Five levels of refinement are used in this case. Through a superposition of such functions, a solution with two peaks at points (0,0) and (0.5,1), is also investigated, using four refinement levels (Table 4.2.1.2.). In all cases, $\nu_1 = 1$ pre–smoothing and $\nu_2 = 1$ post–smoothing are performed on the finer grid levels, while on the coarsest grid $10(\nu_1 + \nu_2)$ smoothings are done. We iterate until the residuals on each grid fall $1/\varepsilon = 10^7$ times. The coarsest grid (level 0) has 10^2 CVs and further each coarse CVs produces 4 fine grid CVs at the next level. This way the finest grid at the fifth level for the first problem would have 320^2 CVs in the case of global refinement. On the coarsest grid the solution is global, i.e. within the whole domain. Then applying the local estimation criteria the adaptive LR algorithm starts working.

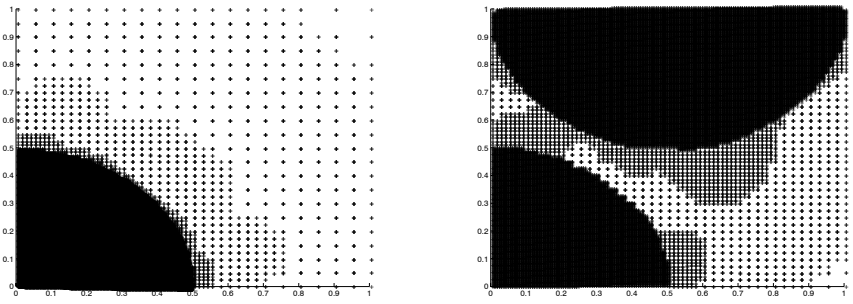

Fig. 4.2.1.1. Adaptive grids for Poisson equation. One-peak function with five refinement levels (left) and two-peaks function, four refinement levels.

For both, single-peak and two-peaks solutions, the indicator (25) connected with the local gradient is used. The value of the free parameter is chosen to be $\gamma = 1e-3$.

This way, the grid configurations represented on Fig. 4.2.1.1. have been obtained. Both tables 4.2.1.1. and 4.2.1.2. compare the solution on a locally refined grid with the standard multigrid solution. It is seen that the MG-ALR solutions have the same accuracy as the MG ones, but use significantly less resources (memory, i.e. active CVs, and CPU time therefore).

4.2.2 Lid-driven flow in a square cavity

As a bench-mark comparison we use [5]. The solver in its standard MG-variant has been extensively tested for such 2D and 3D problems for different Re numbers and very good correspondence with the bench-mark data has been obtained. Computations with the adaptive MG-LR solver in 2D case are discussed below. The following parameters of the iterative MG-LR solver are used: six pre- and post- smoothings (i.e., SIMPLE iterations), $\nu_1 = \nu_2 = 6$, are performed on each grid (except the coarsest one, there we smooth 60 times). The calculations on each grid continue until the residual (vector sum of the velocities C-norms) falls $1/\varepsilon = 10^4$ times. The convective terms in (2) are discretized with central difference scheme, in order to have approximation for the bench-mark variables stream function (ψ) and vorticity (ω), calculated at the post-processing stage.

We do adaptive refinement on the highest one (Re=100) or two (Re=400) grid levels, while on the lower grid levels we use a standard multigrid, i.e. global refinement. It is known, that small vortices appear in the bottom corners of the cavity for moderate Re numbers. Our aim is to use such a combination of criteria which can provide the refinement subdomains on the next grid level(s) to occupy (approximately) the zones of interest for us: (i) the corners of the cavity on the top (where the solution for ω has singular points) and (ii) around the bottom corners, where the secondary vortices appear.

Before to discuss our numerical results for both 2D and 3D cavity let us make a remark about the notation. The first row of each table, presenting the "convergence history" of a particular simulation, shows the grid size - the number of "virtual" CVs (denoted also by "v"), i.e. the CVs that the mesh would have contained in the case of a standard multigrid. It is clear, that the addaptive refinement approach uses less CVs, we call them "actual" CVs, which have been also given in the tables.

Lid-driven flow in a square cavity, Re=100.

In this case as a local sensor we use the Richardson error, calculated for the quantity $(\omega^2+\psi^2)^{1/2}$, where the variables ψ and ω are locally scaled with their coarse-grid values. A certain CV is to be refined, if $e^h_{(\psi,\omega)}(x,y) > 0.3$, i.e. $\gamma_{(\psi,\omega)} = 2.25e-3$ in (25). We apply the LR algorithm on the last, 4-th grid level, the results are presented on Fig. 4.2.2.1. (left) and Table 4.2.2.1.

Table 4.2.2.1. Re=100, one-level adaptive refinement on the finest grid

	grid0 [10,10]	grid1 [20,20]	grid2 [40,40]	grid3 [80,80]v	bench–mark
"actual" CVs	100	400	1600	1904	
num. sweeps	-	2	2	3	
PRMRY ψ_{min}	-9.399817e-2	-1.044355e-1	-1.037122e-1	-1.032615e-1	-0.103423
LB ψ_{max}	4.767103e-5	6.659719e-5	8.305851e-6	5.058273e-6	1.74877e-6
RB ψ_{max}	1.520927e-4	1.191102e-4	2.145679e-5	2.026372e-5	1.25374e-5

Lid-driven flow in a square cavity, Re=400.

The same criteria is used as in the previous case: we refine the CVs, where $e^h_{(\psi,\omega)}(x,y) > 0.2$, i.e. $f_{(\psi,\omega)} = 1.65e-2$ in (25). We adaptively refine the grid on the highest two levels. The results are presented on Fig.4.2.2.1. (right) and Table 4.2.2.2.

Table 4.2.2.2. Re=400, two refinement levels

	grid0 [10,10]	grid1 [20,20]	grid2 [40,40]v	grid3 [80,80]v	bench–mark
"actual" CVs	100	400	1492	4784	
num. sweeps	-	28	27	27	
PRMRY ψ_{min}	-8.087013e-2	-1.081892e-1	-1.091522e-1	-1.120618e-1	-0.113909
LB ψ_{max}	-7.676021e-5	-1.282421e-5	3.046287e-5	3.425609e-5	1.41951e-5
RB ψ_{max}	1.284920e-3	3.313964e-4	6.118039e-4	6.600418e-4	6.42352e-4

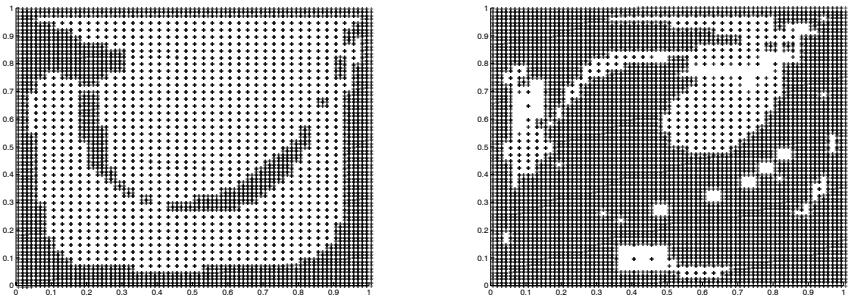

Fig. 4.2.2.1. Composite grids for cavity flow on the finest 4-th level, Re=100 (left) and Re=400.

The tables above show a very good correspondence with the benchmark data for the both cases considered. The same conclusion is valid for the vorticity (ω) benchmark values.

4.3 Adaptive local refinement, 3D case: lid-driven flow in a unit cavity

4.3.1 Refinement criteria.

For the 3D Navier-Stokes equations we employ again criteria based on the Richardson error. On a certain grid level (but at least the second one) we calculate $e^h(x,y,z)$ for

all variables (the velocities u, v, w and pressure p) and scale it locally - i.e. dividing by the value of the corresponding variable in the fine grid CV. Then we (locally) define e^h as

$$e^h(x,y,z) = \left((e_u^h)^2 + (e_v^h)^2 + (e_w^h)^2\right)^{0.5}.$$

The threshold values of the sensor e^h will be given in each particular case of adaptive refinement (AR).

In the following sections first we validate our solver by comparison with benchmark data and then we assess to what extend the characteristic velocity profiles, obtained through adaptive local refinement (i.e. with "cheaper" calculations), approach those, obtained through full multigrid (MG) calculations on the finest grid.

4.3.2 Validation of the MG solver

Table 4.3.2.1. shows a quantitative comparison with the data from [8].

Table 4.3.2.1. 3D cavity, $Re = 1000$: velocity profiles extrema compared with [8]

Scheme	grid size	U_{min}, $x = y = 0.5$	W_{min}, $y = z = 0.5$	W_{max}, $y = z = 0.5$
UPW	16^3	-1.138055e-1	-2.318399e-1	1.003921e-1
	16^3, 10^{-6}	-1.138950e-1	-2.318897e-1	1.005027e-1
	16^3, [8]	-1.2307e-1	-2.4441e-1	1.1413e-1
UPW	32^3	-1.552199e-1	-2.743511e-1	1.444918e-1
	32^3, 10^{-6}	-1.555689e-1	-2.750799e-1	1.445480e-1
	32^3, [8]	-1.7516e-1	-3.1752e-1	1.5902e-1
UPW	64^3	-2.076872e-1	-3.516547e-1	1.865642e-1
	64^3, 10^{-6}	-2.090728e-1	-3.526455e-1	1.890350e-1
	64^3, [8]	-2.1849e-1	-3.7466e-1	1.945e-1
CNTR	32^3	-2.221334e-1	-3.55598e-1	1.972862e-1
	32^3, [8]	-2.5207e-1	-3.9769e-1	2.2159e-1
CNTR	64^3	-2.614485e-1	-4.092114e-1	2.298663e-1
	64^3, [8]	-2.7245e-1	-4.2502e-1	2.4002e-1

It compares the extrema along certain velocity profiles - those for the u-velocity along the central vertical line of the cavity and those of the w-velocity along the central horizontal line, parallel to the x-axis. In all our 3D experiments we use the same MG-parameters as in 2D, i.e. on each grid $\nu_1 = \nu_2 = 6$ pre- and post-smoothing iterations and the calculations continue until the residual (vector sum of the velocities C-norms) falls $1/\varepsilon = 10^4$ times. The few exceptions when $\varepsilon = 10^{-6}$ have been noted in table 4.3.2.1. by adding this value in the column showing the grid size. The discretization scheme used for the convective terms (upwind or central differences) is also shown

in this table. A qualitative correspondence of the velocity profiles, obtained through our solver, with those of [8] can be seen on the graphics in the following sections. But quantitatively, as a rule, our maximums are lower than these of [8], i.e. the same numerical schemes in our implementation possess higher numerical diffusion. Nevertheless, the comparison seems to be quite acceptable for our solver.

The comparison between our AR- and MG-solutions along these velocity profiles we use as a mean to evaluate the proposed AR-approach. From the MG-solution one can obtain both the "aimed" extremum and its location as well, being able then to assess locally the different variants of AR-solutions. In the tables below we present our simulations for different *Re* numbers with different discretization schemes (central or upwind) for the convective terms used.

4.3.3 Lid-driven flow in a cubic cavity, Re=100, CENTRAL

The following three tables represent the convergence history of the MG and ALR solutions:

Table 4.3.3.1. Re=100, 3D cavity, CENTR, Full MG

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	512000
start residual	2.0e+01	9.696443e+00	3.295980e+00	1.580092e+00
end residual	1.790839e-03	7.668303e-04	3.069158e-04	1.452561e-04
num. smooths	25	20	18	18
num. sweeps	-	2	2	2

Table 4.3.3.2. Re=100, 3D cavity, CENTR, two AR-levls from gr 2, $est = 0.2$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	61808	305168
start residual	2.0e+01	9.696443e+00	3.365413e+00	1.567563e+00
end residual	1.790839e-03	7.668303e-04	1.422811e-04	1.340488e-04
num. smooths	25	20	20	18
num. sweeps	-	2	2	2

Table 4.3.3.3. Re=100, 3D cavity, CENTR, single AR-levl on gr 3., $est = 0.1$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	376056
start residual	2.0e+01	9.696443e+00	3.295980e+00	1.569300e+00
end residual	1.790839e-03	7.668303e-04	3.069158e-04	1.176648e-04
num. smooths	25	20	18	19
num. sweeps	-	2	2	2

Fig. 4.3.3.1. shows the finest grids in the central cross-section plane $y = 0.5$, and Fig. 4.3.3.2. compares the characteristic velocity profiles in the MG and ALR cases. Table 4.3.3.4. gives a quantitative assessment about the latter comparison.

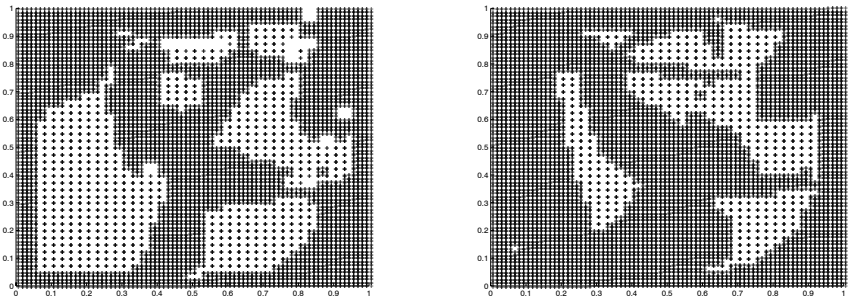

Fig. 4.3.3.1. Adaptive grids at the cross-section y=0.5, 3D cavity, Re=100, CENTRAL. The finest grid for two AR-levels (left) and a single AR-level.

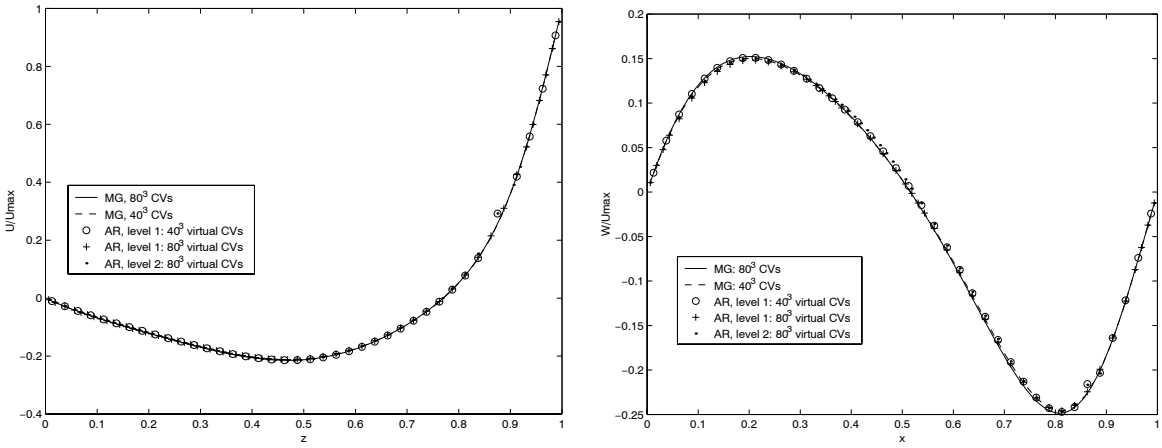

Fig. 4.3.3.2. Re=100, central: U-velocity at x=y=0.5 (left) and W-velocity at y=z=0.5.

Table 4.3.3.4. 3D cavity, $Re = 100$, CENTR. Extrema on velocity profiles

grid size	U_{min}	z	W_{min}	x	W_{max}	x
MG 40^3	-2.113e-1	4.625e-1	-2.462e-1	8.125e-1	1.497e-1	2.125e-1
MG 80^3	-2.144e-1	4.6875e-1	-2.484e-1	8.0625e-1	1.521e-1	2.0625e-1
AR, lev 1, 40^3 (v)	-2.141e-1	4.625e-1	-2.469e-1	8.125e-1	1.509e-1	2.125e-1
AR, lev 2, 80^3 (v)	-2.159e-1	4.6875e-1	-2.464e-1	8.125e-1	1.505e-1	2.125e-1
AR, lev 1, 80^3 (v)	-2.140e-1	4.6875e-1	-2.484e-1	8.125e-1	1.539e-1	2.0625e-1

For this case $Re = 100$ the graphics of the velocity profiles show that the solution is accurate enough even on the coarser grids. But even the fact, that the AR-solutions do not deteriorate the profiles, rather shows the correctness of the AR-approach.

4.3.4 Lid-driven flow in a cubic cavity, Re=400, CENTRAL

The following three tables show the convergence history for the MG and ALR case.

Table 4.3.4.1. Re=400, 3D cavity, CENTRAL, Full MG

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	512000
start residual	8.000000e+01	1.033482e+02	2.954277e+01	8.957048e+00
end residual	7.964146e-03	9.411920e-03	2.911198e-03	8.850487e-04
num. smooths	573	320	311	326
num. sweeps	-	27	26	28

Table: 4.3.4.2. Re=400, 3D cavity, CENTR, two AR-levls from gr 2, $est = 0.3$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	63104	394992
start residual	8.000000e+01	1.033482e+02	3.048354e+01	9.309357e+00
end residual	7.964146e-03	9.411920e-03	2.966908e-03	9.139852e-04
num. smooths	573	320	308	318
num. sweeps	-	27	26	27

Table 4.3.4.3. Re=400, 3D cavity, CENTR, single AR-levl on gr 3, $est = 0.3$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	397504
start residual	8.000000e+01	1.033482e+02	2.954277e+01	9.311269e+00
end residual	7.964146e-03	9.411920e-03	2.911198e-03	9.087581e-04
num. smooths	573	320	311	318
num. sweeps	-	27	26	27

The figures 4.3.4.1. and 4.3.4.2. represent correspondingly the meshes on the finest refinement levels and the comparison of the velocity profiles. The extrema of the latter have been compared also in Table 4.3.4.4.

Table 4.3.4.4. 3D cavity, $Re = 400$, CENTRAL. Extrema on velocity profiles

grid size	U_{min}	z	W_{min}	x	W_{max}	x
MG 40^3	-2.192e-1	2.375e-1	-3.591e-1	8.625e-1	1.911e-1	1.375e-1
MG 80^3	-2.320e-1	2.4375e-1	-3.749e-1	8.6875e-1	2.037e-1	1.4375e-1
AR, lev 1, 40^3 (v)	-2.193e-1	2.375e-1	-3.612e-1	8.625e-1	1.892e-1	1.375e-01
AR, lev 2, 80^3 (v)	-2.304e-1	2.4375e-1	-3.791e-1	8.6875e-1	1.931e-1	1.375e-1
AR, lev 1, 80^3 (v)	-2.304e-1	2.4375e-1	-3.637e-1	8.5625e-1	1.960e-1	1.375e-1

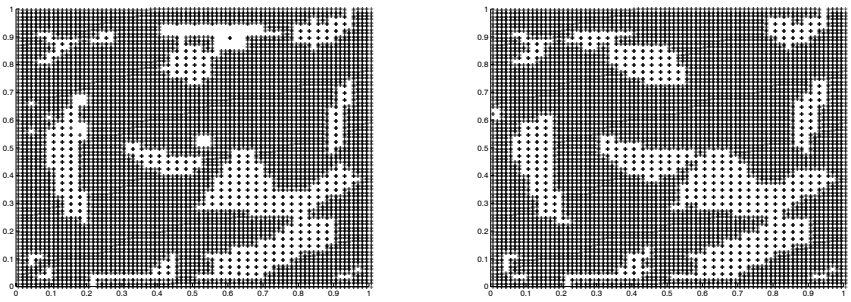

Fig. 4.3.4.1. Adaptive grid at the cross-section y=0.5, 3D cavity, Re=400, CENTRAL. The finest grid for two AR-levels (left) and a single AR-level.

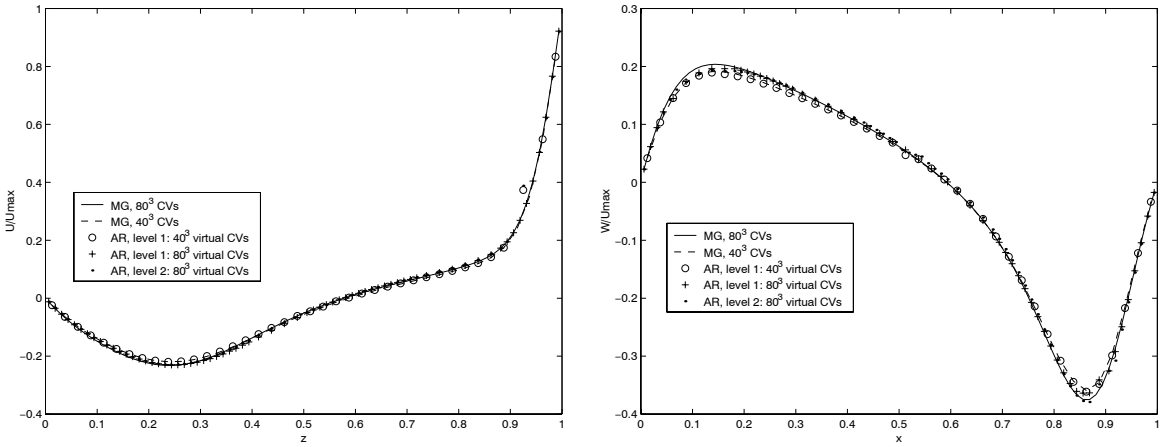

Fig. 4.3.4.2. Re=400, central: U-velocity at x=y=0.5 (left) and W-velocity at y=z=0.5.

For the case $Re = 400$, central, the correspondence with the MG-solution is also good. The values we consider get better even if there is no refinement in the region of interest - see e.g. the values for the W-velocity.

4.3.5 Lid-driven flow in a cubic cavity, Re=400, UPWIND

Again the following three tables represent the convergence history of the MG and ALR solutions. Because the upwind scheme provides a better stability of the iteration method, one can use larger under-relaxation factors and to get a faster convergence in comparison with the central difference discretization. But due to the numerical diffusion, introduced by the upwind scheme, the accuracy gets worse: the velocity extrema significantly decrease by module.

Table 4.3.5.1. Re=400, 3D cavity, UPWIND, Full MG

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	512000
start residual	8.0e+01	1.334946e+02	3.856447e+01	1.129182e+01
end residual	6.751227e-03	1.192929e-02	3.580502e-03	1.090900e-03
num. smooths	28	25	27	25
num. sweeps	-	3	3	3

Table 4.3.5.2. Re=400, 3D cavity, UPW, two AR-levls from gr 2, $est = 0.5$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	57904	275576
start residual	8.0e+01	1.334946e+02	3.933560e+01	1.157837e+01
end residual	6.751227e-03	1.192929e-02	3.686861e-03	1.047248e-03
num. smooths	28	25	24	27
num. sweeps	-	3	2	3

Table 4.3.5.3. Re=400, 3D cavity, UPW, single AR-levl on gr 3, $est = 0.5e$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	303856
start residual	8.0e+01	1.334946e+02	3.856447e+01	1.157649e+01
end residual	6.751227e-03	1.192929e-02	3.580502e-03	9.823030e-04
num. smooths	28	25	27	24
num. sweeps	-	3	3	3

Fig.4.3.5.1. shows the composite meshes on the finest levels and Fig.4.3.5.2. compares the characteristic velocity profiles.

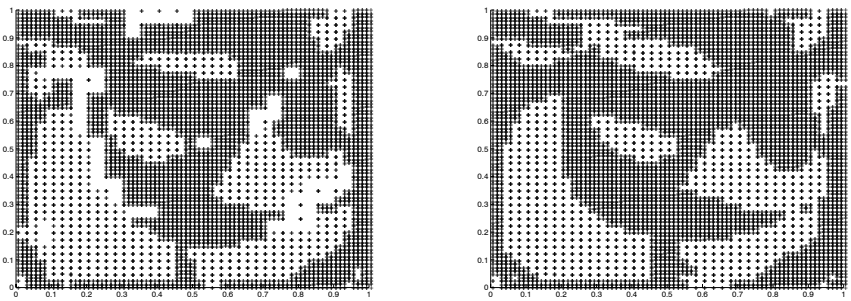

Fig. 4.3.5.1. Adaptive grid at the cross-section y=0.5, 3D cavity, Re=400, UPWIND. The finest grid for two AR-levels (left) and a single AR-level.

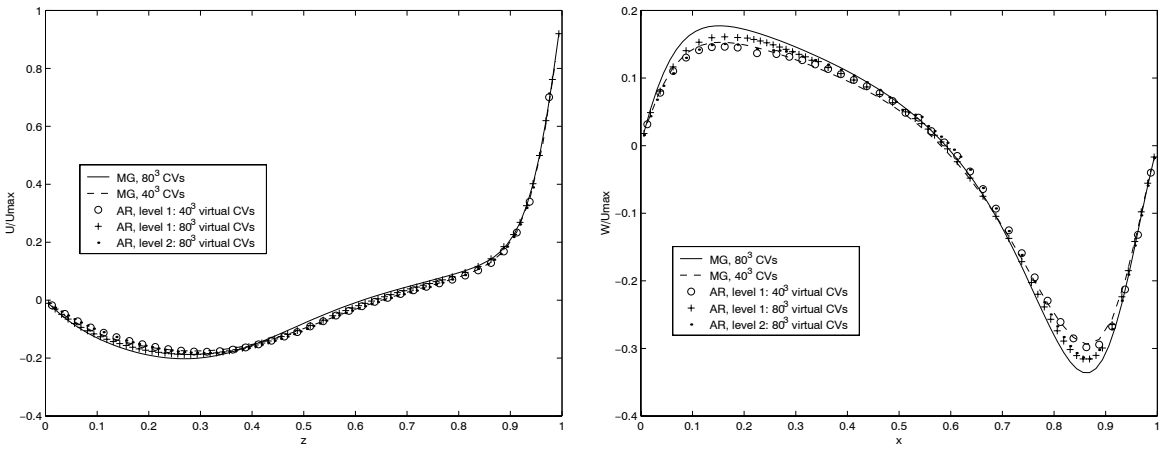

Fig. 4.3.5.2. Re=400, upwind: U-velocity at x=y=0.5 (left) and W-velocity at y=z=0.5.

Table 4.3.5.4. 3D cavity, $Re = 400$, UPWIND. Extrema on velocity profiles

grid size	U_{min}	z	W_{min}	x	W_{max}	x
MG 40^3	-1.755e-1	2.875e-1	-2.937e-1	8.625e-1	1.530e-1	1.625e-1
MG 80^3	-2.023e-1	2.6875e-1	-3.357e-1	8.6875e-1	1.773e-1	1.5625e-1
AR, lev 1, 40^3 (v)	-1.780e-1	3.125e-1	-2.982e-1	8.625e-1	1.462e-1	1.625e-1
AR, lev 2, 80^3 (v)	-1.828e-1	3.0625e-1	-3.153e-1	8.6875e-1	1.503e-1	1.625e-1
AR, lev 1, 80^3 (v)	-1.873e-1	2.8125e-1	-3.155e-1	8.6875e-1	1.609e-1	1.625e-1

It can be seen also from Table 4.3.5.4. that the trend in the velocity extrema has been "caught" by the ALR solutions even if there is no refinement around the corresponding extrema (e.g. W_{min}). But the numerical diffusion of the scheme makes the gap between the MG- and ALR- values noticeable.

4.3.6 Lid-driven flow in a cubic cavity, Re=1000, CENTRAL

Because of the higher Re and the CENTRAL discretization scheme, the coarsest grid here is 20^3. (One needs very low underrelaxation factors.) Therefore, we have single-level AR on the finest grid-level (with 80^3 virtual CVs). The convergence history is shown in the following two tables.

Table 4.3.6.1. Re=1000, 3D cavity, CENTRAL, Full MG

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	512000
start residual	-	1.000000e+02	1.641104e+02	4.395887e+01
end residual	-	9.934925e-03	1.634523e-02	4.286690e-03
num. smooths	-	1804	268	261
num. sweeps	-	-	23	22

Table 4.3.6.2. Re=1000, 3D cavity, CENTR, single AR-levl on gr 3, $est = 0.3$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	476832
start residual	-	1.000000e+02	1.641104e+02	4.526716e+01
end residual	-	9.934925e-03	1.634523e-02	4.475994e-03
num. smooths	-	1804	268	257
num. sweeps	-	-	23	22

The composite mesh on the finest refinement level can be seen on Fig. 4.3.6.1. At a first glance it seems that the mesh is almost everywhere refined, but rather this is the case only in this cross-section. Otherwise the number of active CVs in the ALR case is quite acceptable (see the convergence history tables).

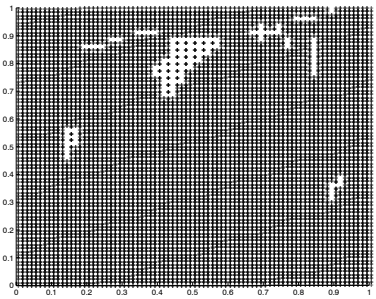

Fig. 4.3.6.1. Adaptive grid at the cross-section y=0.5, 3D cavity, Re=1000, CENTRAL. The finest grid for a single AR-level.

Table 4.3.6.3. 3D cavity, $Re = 1000$, CENTRAL. Extrema on velocity profiles

grid size	U_{min}	z	W_{min}	x	W_{max}	x
MG 40^3	-2.381e-1	1.375e-1	-3.794e-1	9.125e-1	2.109e-1	1.125e-1
MG 80^3	-2.678e-1	1.3125e-1	-4.177e-1	9.0625e-1	2.356e-1	1.0625e-1
AR, lev 1, 80^3 (v)	-2.704e-1	1.0625e-1	-4.313e-1	9.0625e-1	2.31e-1	1.0625e-1

The Fig. 4.3.6.2. and Table 4.3.6.3. compare the velocity profiles. The fit in the MG and ALR case for $Re = 1000$, cental, is acceptable - the ALR-results are very close to those of the finest grid MG-solution.

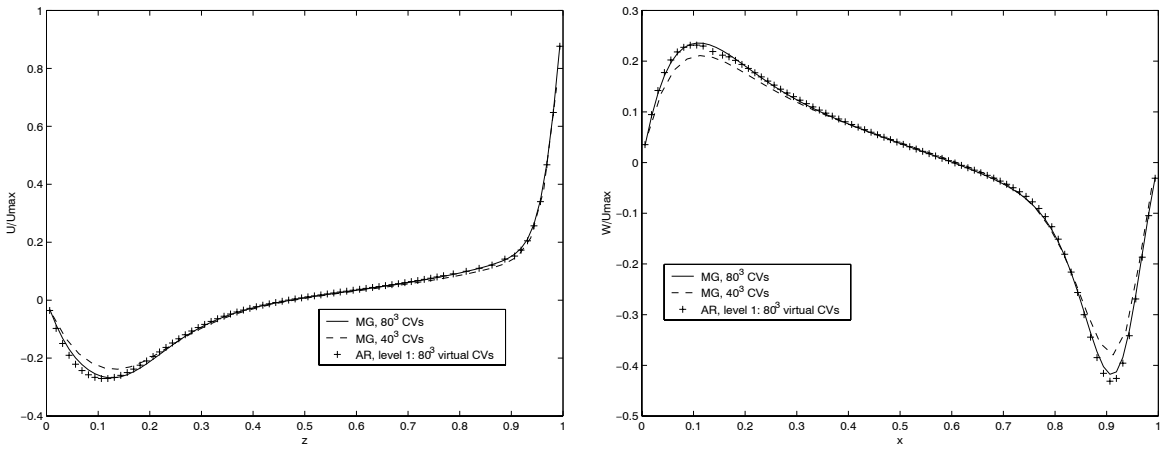

Fig. 4.3.6.2. Re=1000, central: U-velocity at x=y=0.5 (left) and W-velocity at y=z=0.5.

4.3.7 Lid-driven flow in a cubic cavity, Re=1000, UPWIND

The following three tables present the convergence histories of the MG and ALR solutions. Again, because of the stability of the scheme one needs just few MG-sweeps in order to get a convergent solution, but the cost is the lower accuracy of the latter.

Table 4.3.7.1. Re=1000, 3D cavity, UPWIND, Full MG

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	512000
start residual	2.0e+2	7.351936e+02	2.131192e+02	5.859679e+01
end residual	1.847757e-2	6.107434e-02	2.080941e-02	5.652152e-03
num. smooths	32	29	27	25
num. sweeps	-	3	3	3

Table 4.3.7.2. Re=1000, 3D cavity, UPW, two AR-levls from gr 2, $est = 1.0$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	58368	465824
start residual	2.0e+2	7.351936e+02	2.160610e+02	5.960902e+01
end residual	1.847757e-2	6.107434e-02	2.129068e-02	5.674853e-03
num. smooths	32	29	29	26
num. sweeps	-	3	3	3

Table 4.3.7.3. Re=1000, 3D cavity, UPW, single AR-levl on gr 3, $est = 3.0$

	grid 0: 10^3	grid 1: 20^3	grid 2: 40^3	grid 3: 80^3
"actual" CVs	1000	8000	64000	392232
start residual	2.0e+2	7.351936e+02	2.131192e+02	5.972394e+01
end residual	1.847757e-2	6.107434e-02	2.080941e-02	5.899230e-03
num. smooths	32	29	27	32
num. sweeps	-	3	3	3

Fig. 4.3.7.1. shows the composite grids on the finest refinement levels in the cross-section plane and Fig. 4.3.7.2. compares the characteristic velocity profiles.

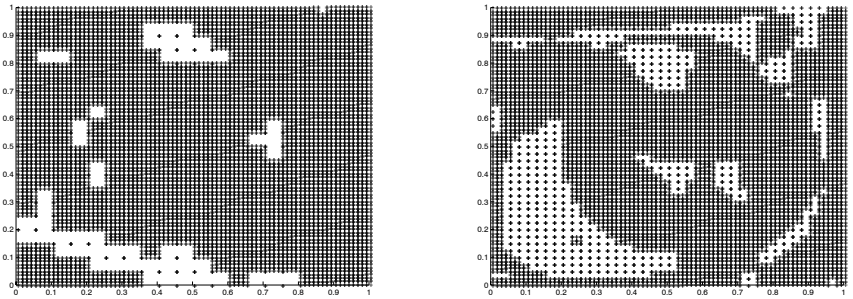

Fig. 4.3.7.1. Adaptive grid at the cross-section y=0.5, 3D cavity, Re=1000, UPWIND. The finest grid for two AR-levels (left) and a single AR-level.

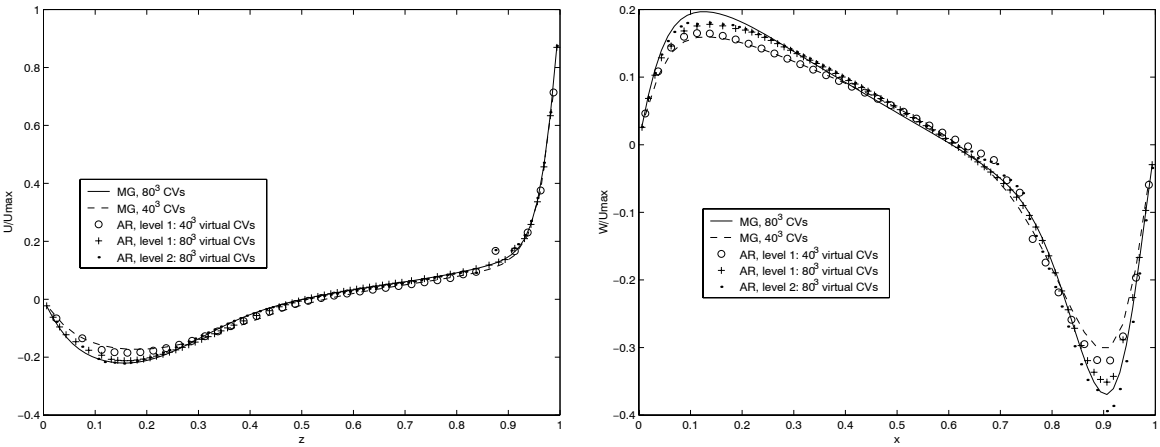

Fig. 4.3.7.2. Re=1000, upwind: U-velocity at x=y=0.5 (left) and W-velocity at y=z=0.5.

Table 4.3.7.4. 3D cavity, $Re = 1000$, UPWIND. Extrema on velocity profiles

grid size	U_{min}	z	W_{min}	x	W_{max}	x
MG 40^3	-1.728e-1	1.875e-1	-3.001e-1	9.125e-1	1.596e-1	1.375e-1
MG 80^3	-2.204e-1	1.5625e-1	-3.695e-1	9.0625e-1	1.967e-1	1.3125e-1
AR, lev 1, 40^3 (v)	-1.852e-1	1.625e-1	-3.252e-1	9.125e-1	1.607e-1	1.125e-1
AR, lev 2, 80^3 (v)	-2.233-1	1.5625e-1	-3.892e-1	9.0625e-1	1.857e-1	1.1875e-1
AR, lev 1, 80^3 (v)	-2.122e-1	1.5625e-1	-3.511e-1	9.0625e-1	1.780e-1	1.375e-1

The results for $Re = 1000$, upwind, are also acceptable. Especially around the extrema, the AR-solution approaches near the MG-one. Only the location of w_{max} is a bit shifted. Of course, because the numerical diffusion, the extrema have been "smoothed", but the trend is clearly seen.

5 Conclusion

A MG solver with adaptive local grid refinement for incompressible Navier-Stokes equations is presented. The basic principles of its software design are also discussed. Error indicators are used to construct adaptively the refinement subdomain. The solver has been tested and verified for 2D Poisson equation and for a lid–driven cavity flow. The adaptive local refinement extension of the multigrid solver allows for a significant reduction of the arithmetic operations and memory (compared with the standard MG case) needed a certain accuracy of the numerical solution to be achieved. At the same time, the optimal multigrid convergence rate is also achieved.

The proposed AR-approach has been shown to work in the 3D case also. As it might be expected, when one refines in the subdomain of interest, the solution there approaches the MG-solution. But in general, even if one does not refine the grid in the region of interest, the solution there becomes better - this can be seen in some of the examples here. It seems, that (in most of the cases) two levels of refinement, with smaller number of "active" CVs on the finest level, give an approximation good enough to the MG-solution. Often this solution seems to be better than the one obtained by single AR-level on the finest grid, even with more "active" CVs there.

Of course, this statement can not be generalized: the question how to chose the subdomain of interest, where to refine the grid, is essentially open, i.e. the question of the refinement criteria.

References

[1] W.Chen, F.Lien, M,Leschziner, Local mesh refinement within a multiblock structured-grid scheme for general flows, *Comput. Methods Appl. Mech. Eng.* **144**, 327 (1997).

[2] R.Ewing, R.Lazarov, P.Vassilevski, Local refinement techniques for elliptic problems on cell–centered grids, II: optimal two–grid iterative methods, *Num. Linear Algebra and Applications*, **1**, No.4, pp 337-368 (1994).

[3] J.H.Ferziger, M.Peric, *Computational methods for fluid dynamics* (Springer, 1999).

[4] C.A.J.Fletcher, *Computational Techniques for Fluid Dynamics*, (Springer-Verlag, 1991).

[5] 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** pp. 387-411 (1982).

[6] W.Hackbusch, *Multi-Grid Methods and Applications*, (Springer-Verlag, 1985).

[7] O.Iliev, D.Stoyanov, On a multigrid, local refinement solver for incompressible Navier-Stokes equations, to appear in *Mathematical Modelling*, **13**, No.8 (2001).

[8] Lilek, Z., Muzaferija S., Perić M., Efficiency and accuracy aspects of a full-multigrid SIMPLE algorithm for three-dimensional flows, *Numer. Heat Transfer, Part B*, **31**, pp. 23-42 (1997).

[9] Stephen F. McCormick, *Multilevel Adaptive Methods for Partial Differential Equations*, (SIAM, Philadelphia, 1989).

[10] S.V.Patankar, *Numerical Heat Transfer and Fluid Flow*, (Hemisphere, 1980).

[11] M.Perić, R.Kessler and G.Scheuerer, Comparison on finite-volume numerical methods with staggered and collocated grids, *Comput. Fluids* **16** pp. 389-403 (1988).

[12] A.A.Samarskii, *Theory of difference schemes*, (Moskow, 1977, in Russian).

[13] Turek S., *Efficient solvers for incompressible flow problems: An algorithmic approach in view of computational aspects*, (Springer, 1998/99).

[14] R. Verfuerth, *A review of a posteriori error estimation and adaptive mesh refinement techniques*, (Wiley-Teubner, Chichester, Stutgard, 1996).

[15] P.Wesseling, *An introduction to multigrid methods*, (Wiley, Chichester, 1992).

Published reports of the Fraunhofer ITWM

The PDF-files of the following reports are available under:
www.itwm.fraunhofer.de/rd/presse/berichte

1. D. Hietel, K. Steiner, J. Struckmeier

A Finite - Volume Particle Method for Compressible Flows

We derive a new class of particle methods for conservation laws, which are based on numerical flux functions to model the interactions between moving particles. The derivation is similar to that of classical Finite-Volume methods; except that the fixed grid structure in the Finite-Volume method is substituted by so-called mass packets of particles. We give some numerical results on a shock wave solution for Burgers equation as well as the well-known one-dimensional shock tube problem.
(19 pages, 1998)

2. M. Feldmann, S. Seibold

Damage Diagnosis of Rotors: Application of Hilbert Transform and Multi-Hypothesis Testing

In this paper, a combined approach to damage diagnosis of rotors is proposed. The intention is to employ signal-based as well as model-based procedures for an improved detection of size and location of the damage. In a first step, Hilbert transform signal processing techniques allow for a computation of the signal envelope and the instantaneous frequency, so that various types of non-linearities due to a damage may be identified and classified based on measured response data. In a second step, a multi-hypothesis bank of Kalman Filters is employed for the detection of the size and location of the damage based on the information of the type of damage provided by the results of the Hilbert transform.
Keywords: Hilbert transform, damage diagnosis, Kalman filtering, non-linear dynamics
(23 pages, 1998)

3. Y. Ben-Haim, S. Seibold

Robust Reliability of Diagnostic Multi-Hypothesis Algorithms: Application to Rotating Machinery

Damage diagnosis based on a bank of Kalman filters, each one conditioned on a specific hypothesized system condition, is a well recognized and powerful diagnostic tool. This multi-hypothesis approach can be applied to a wide range of damage conditions. In this paper, we will focus on the diagnosis of cracks in rotating machinery. The question we address is: how to optimize the multi-hypothesis algorithm with respect to the uncertainty of the spatial form and location of cracks and their resulting dynamic effects. First, we formulate a measure of the reliability of the diagnostic algorithm, and then we discuss modifications of the diagnostic algorithm for the maximization of the reliability. The reliability of a diagnostic algorithm is measured by the amount of uncertainty consistent with no-failure of the diagnosis. Uncertainty is quantitatively represented with convex models.
Keywords: Robust reliability, convex models, Kalman filtering, multi-hypothesis diagnosis, rotating machinery, crack diagnosis
(24 pages, 1998)

4. F.-Th. Lentes, N. Siedow

Three-dimensional Radiative Heat Transfer in Glass Cooling Processes

For the numerical simulation of 3D radiative heat transfer in glasses and glass melts, practically applicable mathematical methods are needed to handle such problems optimal using workstation class computers. Since the exact solution would require super-computer capabilities we concentrate on approximate solutions with a high degree of accuracy. The following approaches are studied: 3D diffusion approximations and 3D ray-tracing methods.
(23 pages, 1998)

5. A. Klar, R. Wegener

A hierarchy of models for multilane vehicular traffic
Part I: Modeling

In the present paper multilane models for vehicular traffic are considered. A microscopic multilane model based on reaction thresholds is developed. Based on this model an Enskog like kinetic model is developed. In particular, care is taken to incorporate the correlations between the vehicles. From the kinetic model a fluid dynamic model is derived. The macroscopic coefficients are deduced from the underlying kinetic model. Numerical simulations are presented for all three levels of description in [10]. Moreover, a comparison of the results is given there.
(23 pages, 1998)

Part II: Numerical and stochastic investigations

In this paper the work presented in [6] is continued. The present paper contains detailed numerical investigations of the models developed there. A numerical method to treat the kinetic equations obtained in [6] are presented and results of the simulations are shown. Moreover, the stochastic correlation model used in [6] is described and investigated in more detail.
(17 pages, 1998)

6. A. Klar, N. Siedow

Boundary Layers and Domain Decomposition for Radiative Heat Transfer and Diffusion Equations: Applications to Glass Manufacturing Processes

In this paper domain decomposition methods for radiative transfer problems including conductive heat transfer are treated. The paper focuses on semi-transparent materials, like glass, and the associated conditions at the interface between the materials. Using asymptotic analysis we derive conditions for the coupling of the radiative transfer equations and a diffusion approximation. Several test cases are treated and a problem appearing in glass manufacturing processes is computed. The results clearly show the advantages of a domain decomposition approach. Accuracy equivalent to the solution of the global radiative transfer solution is achieved, whereas computation time is strongly reduced.
(24 pages, 1998)

7. I. Choquet

Heterogeneous catalysis modelling and numerical simulation in rarified gas flows Part I: Coverage locally at equilibrium

A new approach is proposed to model and simulate numerically heterogeneous catalysis in rarefied gas flows. It is developed to satisfy all together the following points:
1) describe the gas phase at the microscopic scale, as required in rarefied flows,
2) describe the wall at the macroscopic scale, to avoid prohibitive computational costs and consider not only crystalline but also amorphous surfaces,
3) reproduce on average macroscopic laws correlated with experimental results and
4) derive analytic models in a systematic and exact way. The problem is stated in the general framework of a non static flow in the vicinity of a catalytic and non porous surface (without aging). It is shown that the exact and systematic resolution method based on the Laplace transform, introduced previously by the author to model collisions in the gas phase, can be extended to the present problem. The proposed approach is applied to the modelling of the EleyRideal and LangmuirHinshelwood recombinations, assuming that the coverage is locally at equilibrium. The models are developed considering one atomic species and extended to the general case of several atomic species. Numerical calculations show that the models derived in this way reproduce with accuracy behaviors observed experimentally.
(24 pages, 1998)

8. J. Ohser, B. Steinbach, C. Lang

Efficient Texture Analysis of Binary Images

A new method of determining some characteristics of binary images is proposed based on a special linear filtering. This technique enables the estimation of the area fraction, the specific line length, and the specific integral of curvature. Furthermore, the specific length of the total projection is obtained, which gives detailed information about the texture of the image. The influence of lateral and directional resolution depending on the size of the applied filter mask is discussed in detail. The technique includes a method of increasing directional resolution for texture analysis while keeping lateral resolution as high as possible.
(17 pages, 1998)

9. J. Orlik

Homogenization for viscoelasticity of the integral type with aging and shrinkage

A multiphase composite with periodic distributed inclusions with a smooth boundary is considered in this contribution. The composite component materials are supposed to be linear viscoelastic and aging (of the nonconvolution integral type, for which the Laplace transform with respect to time is not effectively applicable) and are subjected to isotropic shrinkage. The free shrinkage deformation can be considered as a fictitious temperature deformation in the behavior law. The procedure presented in this paper proposes a way to determine average (effective homogenized) viscoelastic and shrinkage (temperature) composite properties and the homogenized stressfield from known properties of the components. This is done by the extension of the asymptotic homogenization technique known for pure elastic nonhomogeneous bodies to the nonhomogeneous thermoviscoelasticity of the integral noncon-

volution type. Up to now, the homogenization theory has not covered viscoelasticity of the integral type. SanchezPalencia (1980), Francfort & Suquet (1987) (see [2], [9]) have considered homogenization for viscoelasticity of the differential form and only up to the first derivative order. The integralmodeled viscoelasticity is more general then the differential one and includes almost all known differential models. The homogenization procedure is based on the construction of an asymptotic solution with respect to a period of the composite structure. This reduces the original problem to some auxiliary boundary value problems of elasticity and viscoelasticity on the unit periodic cell, of the same type as the original non-homogeneous problem. The existence and uniqueness results for such problems were obtained for kernels satisfying some constrain conditions. This is done by the extension of the Volterra integral operator theory to the Volterra operators with respect to the time, whose 1 kernels are space linear operators for any fixed time variables. Some ideas of such approach were proposed in [11] and [12], where the Volterra operators with kernels depending additionally on parameter were considered. This manuscript delivers results of the same nature for the case of the spaceoperator kernels.
(20 pages, 1998)

10. J. Mohring

Helmholtz Resonators with Large Aperture

The lowest resonant frequency of a cavity resonator is usually approximated by the classical Helmholtz formula. However, if the opening is rather large and the front wall is narrow this formula is no longer valid. Here we present a correction which is of third order in the ratio of the diameters of aperture and cavity. In addition to the high accuracy it allows to estimate the damping due to radiation. The result is found by applying the method of matched asymptotic expansions. The correction contains form factors describing the shapes of opening and cavity. They are computed for a number of standard geometries. Results are compared with numerical computations.
(21 pages, 1998)

11. H. W. Hamacher, A. Schöbel

On Center Cycles in Grid Graphs

Finding "good" cycles in graphs is a problem of great interest in graph theory as well as in locational analysis. We show that the center and median problems are NP hard in general graphs. This result holds both for the variable cardinality case (i.e. all cycles of the graph are considered) and the fixed cardinality case (i.e. only cycles with a given cardinality p are feasible). Hence it is of interest to investigate special cases where the problem is solvable in polynomial time. In grid graphs, the variable cardinality case is, for instance, trivially solvable if the shape of the cycle can be chosen freely. If the shape is fixed to be a rectangle one can analyze rectangles in grid graphs with, in sequence, fixed dimension, fixed cardinality, and variable cardinality. In all cases a complete characterization of the optimal cycles and closed form expressions of the optimal objective values are given, yielding polynomial time algorithms for all cases of center rectangle problems. Finally, it is shown that center cycles can be chosen as rectangles for small cardinalities such that the center cycle problem in grid graphs is in these cases completely solved.
(15 pages, 1998)

12. H. W. Hamacher, K.-H. Küfer

Inverse radiation therapy planning - a multiple objective optimisation approach

For some decades radiation therapy has been proved successful in cancer treatment. It is the major task of clinical radiation treatment planning to realize on the one hand a high level dose of radiation in the cancer tissue in order to obtain maximum tumor control. On the other hand it is obvious that it is absolutely necessary to keep in the tissue outside the tumor, particularly in organs at risk, the unavoidable radiation as low as possible.
No doubt, these two objectives of treatment planning - high level dose in the tumor, low radiation outside the tumor - have a basically contradictory nature. Therefore, it is no surprise that inverse mathematical models with dose distribution bounds tend to be infeasible in most cases. Thus, there is need for approximations compromising between overdosing the organs at risk and underdosing the target volume.
Differing from the currently used time consuming iterative approach, which measures deviation from an ideal (non-achievable) treatment plan using recursively trial-and-error weights for the organs of interest, we go a new way trying to avoid a priori weight choices and consider the treatment planning problem as a multiple objective linear programming problem: with each organ of interest, target tissue as well as organs at risk, we associate an objective function measuring the maximal deviation from the prescribed doses.
We build up a data base of relatively few efficient solutions representing and approximating the variety of Pareto solutions of the multiple objective linear programming problem. This data base can be easily scanned by physicians looking for an adequate treatment plan with the aid of an appropriate online tool.
(14 pages, 1999)

13. C. Lang, J. Ohser, R. Hilfer

On the Analysis of Spatial Binary Images

This paper deals with the characterization of microscopically heterogeneous, but macroscopically homogeneous spatial structures. A new method is presented which is strictly based on integral-geometric formulae such as Crofton's intersection formulae and Hadwiger's recursive definition of the Euler number. The corresponding algorithms have clear advantages over other techniques. As an example of application we consider the analysis of spatial digital images produced by means of Computer Assisted Tomography.
(20 pages, 1999)

14. M. Junk

On the Construction of Discrete Equilibrium Distributions for Kinetic Schemes

A general approach to the construction of discrete equilibrium distributions is presented. Such distribution functions can be used to set up Kinetic Schemes as well as Lattice Boltzmann methods. The general principles are also applied to the construction of Chapman Enskog distributions which are used in Kinetic Schemes for compressible Navier-Stokes equations.
(24 pages, 1999)

15. M. Junk, S. V. Raghurame Rao

A new discrete velocity method for Navier-Stokes equations

The relation between the Lattice Boltzmann Method, which has recently become popular, and the Kinetic Schemes, which are routinely used in Computational Fluid Dynamics, is explored. A new discrete velocity model for the numerical solution of Navier-Stokes equations for incompressible fluid flow is presented by combining both the approaches. The new scheme can be interpreted as a pseudo-compressibility method and, for a particular choice of parameters, this interpretation carries over to the Lattice Boltzmann Method.
(20 pages, 1999)

16. H. Neunzert

Mathematics as a Key to Key Technologies

The main part of this paper will consist of examples, how mathematics really helps to solve industrial problems; these examples are taken from our Institute for Industrial Mathematics, from research in the Technomathematics group at my university, but also from ECMI groups and a company called TecMath, which originated 10 years ago from my university group and has already a very successful history.
(39 pages (4 PDF-Files), 1999)

17. J. Ohser, K. Sandau

Considerations about the Estimation of the Size Distribution in Wicksell's Corpuscle Problem

Wicksell's corpuscle problem deals with the estimation of the size distribution of a population of particles, all having the same shape, using a lower dimensional sampling probe. This problem was originary formulated for particle systems occurring in life sciences but its solution is of actual and increasing interest in materials science. From a mathematical point of view, Wicksell's problem is an inverse problem where the interesting size distribution is the unknown part of a Volterra equation. The problem is often regarded ill-posed, because the structure of the integrand implies unstable numerical solutions. The accuracy of the numerical solutions is considered here using the condition number, which allows to compare different numerical methods with different (equidistant) class sizes and which indicates, as one result, that a finite section thickness of the probe reduces the numerical problems. Furthermore, the relative error of estimation is computed which can be split into two parts. One part consists of the relative discretization error that increases for increasing class size, and the second part is related to the relative statistical error which increases with decreasing class size. For both parts, upper bounds can be given and the sum of them indicates an optimal class width depending on some specific constants.
(18 pages, 1999)

18. E. Carrizosa, H. W. Hamacher, R. Klein, S. Nickel

Solving nonconvex planar location problems by finite dominating sets

It is well-known that some of the classical location problems with polyhedral gauges can be solved in polynomial time by finding a finite dominating set, i.e. a finite set of candidates guaranteed to contain at least one optimal location.
In this paper it is first established that this result holds

for a much larger class of problems than currently considered in the literature. The model for which this result can be proven includes, for instance, location problems with attraction and repulsion, and location-allocation problems.
Next, it is shown that the approximation of general gauges by polyhedral ones in the objective function of our general model can be analyzed with regard to the subsequent error in the optimal objective value. For the approximation problem two different approaches are described, the sandwich procedure and the greedy algorithm. Both of these approaches lead - for fixed epsilon - to polynomial approximation algorithms with accuracy epsilon for solving the general model considered in this paper.
Keywords: Continuous Location, Polyhedral Gauges, Finite Dominating Sets, Approximation, Sandwich Algorithm, Greedy Algorithm
(19 pages, 2000)

19. A. Becker

A Review on Image Distortion Measures

Within this paper we review image distortion measures. A distortion measure is a criterion that assigns a "quality number" to an image. We distinguish between mathematical distortion measures and those distortion measures in-cooperating a priori knowledge about the imaging devices (e. g. satellite images), image processing algorithms or the human physiology. We will consider representative examples of different kinds of distortion measures and are going to discuss them.
Keywords: Distortion measure, human visual system
(26 pages, 2000)

20. H. W. Hamacher, M. Labbé, S. Nickel, T. Sonneborn

Polyhedral Properties of the Uncapacitated Multiple Allocation Hub Location Problem

We examine the feasibility polyhedron of the uncapacitated hub location problem (UHL) with multiple allocation, which has applications in the fields of air passenger and cargo transportation, telecommunication and postal delivery services. In particular we determine the dimension and derive some classes of facets of this polyhedron. We develop some general rules about lifting facets from the uncapacitated facility location (UFL) for UHL and projecting facets from UHL to UFL. By applying these rules we get a new class of facets for UHL which dominates the inequalities in the original formulation. Thus we get a new formulation of UHL whose constraints are all facet–defining. We show its superior computational performance by benchmarking it on a well known data set.
Keywords: integer programming, hub location, facility location, valid inequalities, facets, branch and cut
(21 pages, 2000)

21. H. W. Hamacher, A. Schöbel

Design of Zone Tariff Systems in Public Transportation

Given a public transportation system represented by its stops and direct connections between stops, we consider two problems dealing with the prices for the customers: The fare problem in which subsets of stops are already aggregated to zones and "good" tariffs have to be found in the existing zone system. Closed form solutions for the fare problem are presented for three objective functions. In the zone problem the design of the zones is part of the problem. This problem is NP hard and we therefore propose three heuristics which prove to be very successful in the redesign of one of Germany's transportation systems.
(30 pages, 2001)

22. D. Hietel, M. Junk, R. Keck, D. Teleaga:

The Finite-Volume-Particle Method for Conservation Laws

In the Finite-Volume-Particle Method (FVPM), the weak formulation of a hyperbolic conservation law is discretized by restricting it to a discrete set of test functions. In contrast to the usual Finite-Volume approach, the test functions are not taken as characteristic functions of the control volumes in a spatial grid, but are chosen from a partition of unity with smooth and overlapping partition functions (the particles), which can even move along pre- scribed velocity fields. The information exchange between particles is based on standard numerical flux functions. Geometrical information, similar to the surface area of the cell faces in the Finite-Volume Method and the corresponding normal directions are given as integral quantities of the partition functions. After a brief derivation of the Finite-Volume-Particle Method, this work focuses on the role of the geometric coefficients in the scheme.
(16 pages, 2001)

23. T. Bender, H. Hennes, J. Kalcsics, M. T. Melo, S. Nickel

Location Software and Interface with GIS and Supply Chain Management

The objective of this paper is to bridge the gap between location theory and practice. To meet this objective focus is given to the development of software capable of addressing the different needs of a wide group of users. There is a very active community on location theory encompassing many research fields such as operations research, computer science, mathematics, engineering, geography, economics and marketing. As a result, people working on facility location problems have a very diverse background and also different needs regarding the software to solve these problems. For those interested in non-commercial applications (e. g. students and researchers), the library of location algorithms (LoLA can be of considerable assistance. LoLA contains a collection of efficient algorithms for solving planar, network and discrete facility location problems. In this paper, a detailed description of the functionality of LoLA is presented. In the fields of geography and marketing, for instance, solving facility location problems requires using large amounts of demographic data. Hence, members of these groups (e. g. urban planners and sales managers) often work with geographical information too s. To address the specific needs of these users, LoLA was inked to a geographical information system (GIS) and the details of the combined functionality are described in the paper. Finally, there is a wide group of practitioners who need to solve large problems and require special purpose software with a good data interface. Many of such users can be found, for example, in the area of supply chain management (SCM). Logistics activities involved in strategic SCM include, among others, facility location planning. In this paper, the development of a commercial location software tool is also described. The too is embedded in the Advanced Planner and Optimizer SCM software developed by SAP AG, Walldorf, Germany. The paper ends with some conclusions and an outlook to future activities.
Keywords: facility location, software development, geographical information systems, supply chain management.
(48 pages, 2001)

24. H. W. Hamacher, S. A. Tjandra

Mathematical Modelling of Evacuation Problems: A State of Art

This paper details models and algorithms which can be applied to evacuation problems. While it concentrates on building evacuation many of the results are applicable also to regional evacuation. All models consider the time as main parameter, where the travel time between components of the building is part of the input and the overall evacuation time is the output. The paper distinguishes between macroscopic and microscopic evacuation models both of which are able to capture the evacuees' movement over time.
Macroscopic models are mainly used to produce good lower bounds for the evacuation time and do not consider any individual behavior during the emergency situation. These bounds can be used to analyze existing buildings or help in the design phase of planning a building. Macroscopic approaches which are based on dynamic network flow models (minimum cost dynamic flow, maximum dynamic flow, universal maximum flow, quickest path and quickest flow) are described. A special feature of the presented approach is the fact, that travel times of evacuees are not restricted to be constant, but may be density dependent. Using multicriteria optimization priority regions and blockage due to fire or smoke may be considered. It is shown how the modelling can be done using time parameter either as discrete or continuous parameter.
Microscopic models are able to model the individual evacuee's characteristics and the interaction among evacuees which influence their movement. Due to the corresponding huge amount of data one uses simulation approaches. Some probabilistic laws for individual evacuee's movement are presented. Moreover ideas to model the evacuee's movement using cellular automata (CA) and resulting software are presented. In this paper we will focus on macroscopic models and only summarize some of the results of the microscopic approach. While most of the results are applicable to general evacuation situations, we concentrate on building evacuation.
(44 pages, 2001)

25. J. Kuhnert, S. Tiwari

Grid free method for solving the Poisson equation

A Grid free method for solving the Poisson equation is presented. This is an iterative method. The method is based on the weighted least squares approximation in which the Poisson equation is enforced to be satisfied in every iterations. The boundary conditions can also be enforced in the iteration process. This is a local approximation procedure. The Dirichlet, Neumann and mixed boundary value problems on a unit square are presented and the analytical solutions are compared with the exact solutions. Both solutions matched perfectly.
Keywords: Poisson equation, Least squares method, Grid free method
(19 pages, 2001)

26. T. Götz, H. Rave, D. Reinel-Bitzer, K. Steiner, H. Tiemeier

Simulation of the fiber spinning process

To simulate the influence of process parameters to the melt spinning process a fiber model is used and coupled with CFD calculations of the quench air flow. In the fiber model energy, momentum and mass balance are solved for the polymer mass flow. To calculate the quench air the Lattice Boltzmann method is used. Simulations and experiments for different process parameters and hole configurations are compared and show a good agreement.

Keywords: Melt spinning, fiber model, Lattice Boltzmann, CFD

(19 pages, 2001)

27. A. Zemitis

On interaction of a liquid film with an obstacle

In this paper mathematical models for liquid films generated by impinging jets are discussed. Attention is stressed to the interaction of the liquid film with some obstacle. S. G. Taylor [Proc. R. Soc. London Ser. A 253, 313 (1959)] found that the liquid film generated by impinging jets is very sensitive to properties of the wire which was used as an obstacle. The aim of this presentation is to propose a modification of the Taylor's model, which allows to simulate the film shape in cases, when the angle between jets is different from 180°. Numerical results obtained by discussed models give two different shapes of the liquid film similar as in Taylors experiments. These two shapes depend on the regime: either droplets are produced close to the obstacle or not. The difference between two regimes becomes larger if the angle between jets decreases. Existence of such two regimes can be very essential for some applications of impinging jets, if the generated liquid film can have a contact with obstacles.

Keywords: impinging jets, liquid film, models, numerical solution, shape

(22 pages, 2001)

28. I. Ginzburg, K. Steiner

Free surface lattice-Boltzmann method to model the filling of expanding cavities by Bingham Fluids

The filling process of viscoplastic metal alloys and plastics in expanding cavities is modelled using the lattice Boltzmann method in two and three dimensions. These models combine the regularized Bingham model for viscoplastic with a free-interface algorithm. The latter is based on a modified immiscible lattice Boltzmann model in which one species is the fluid and the other one is considered as vacuum. The boundary conditions at the curved liquid-vacuum interface are met without any geometrical front reconstruction from a first-order Chapman-Enskog expansion. The numerical results obtained with these models are found in good agreement with available theoretical and numerical analysis.

Keywords: Generalized LBE, free-surface phenomena, interface boundary conditions, filling processes, Bingham viscoplastic model, regularized models

(22 pages, 2001)

29. H. Neunzert

»Denn nichts ist für den Menschen als Menschen etwas wert, was er nicht mit Leidenschaft tun kann«

Vortrag anlässlich der Verleihung des Akademiepreises des Landes Rheinland-Pfalz am 21.11.2001

Was macht einen guten Hochschullehrer aus? Auf diese Frage gibt es sicher viele verschiedene, fachbezogene Antworten, aber auch ein paar allgemeine Gesichtspunkte: es bedarf der »Leidenschaft« für die Forschung (Max Weber), aus der dann auch die Begeisterung für die Lehre erwächst. Forschung und Lehre gehören zusammen, um die Wissenschaft als lebendiges Tun vermitteln zu können. Der Vortrag gibt Beispiele dafür, wie in angewandter Mathematik Forschungsaufgaben aus praktischen Alltagsproblemstellungen erwachsen, die in die Lehre auf verschiedenen Stufen (Gymnasium bis Graduiertenkolleg) einfließen; er leitet damit auch zu einem aktuellen Forschungsgebiet, der Mehrskalenanalyse mit ihren vielfältigen Anwendungen in Bildverarbeitung, Materialentwicklung und Strömungsmechanik über, was aber nur kurz gestreift wird. Mathematik erscheint hier als eine moderne Schlüsseltechnologie, die aber auch enge Beziehungen zu den Geistes- und Sozialwissenschaften hat.

Keywords: Lehre, Forschung, angewandte Mathematik, Mehrskalenanalyse, Strömungsmechanik

(18 pages, 2001)

30. J. Kuhnert, S. Tiwari

Finite pointset method based on the projection method for simulations of the incompressible Navier-Stokes equations

A Lagrangian particle scheme is applied to the projection method for the incompressible Navier-Stokes equations. The approximation of spatial derivatives is obtained by the weighted least squares method. The pressure Poisson equation is solved by a local iterative procedure with the help of the least squares method. Numerical tests are performed for two dimensional cases. The Couette flow, Poiseuelle flow, decaying shear flow and the driven cavity flow are presented. The numerical solutions are obtained for stationary as well as instationary cases and are compared with the analytical solutions for channel flows. Finally, the driven cavity in a unit square is considered and the stationary solution obtained from this scheme is compared with that from the finite element method.

Keywords: Incompressible Navier-Stokes equations, Meshfree method, Projection method, Particle scheme, Least squares approximation

AMS subject classification: 76D05, 76M28

(25 pages, 2001)

31. R. Korn, M. Krekel

Optimal Portfolios with Fixed Consumption or Income Streams

We consider some portfolio optimisation problems where either the investor has a desire for an a priori specified consumption stream or/and follows a deterministic pay in scheme while also trying to maximize expected utility from final wealth. We derive explicit closed form solutions for continuous and discrete monetary streams. The mathematical method used is classical stochastic control theory.

Keywords: Portfolio optimisation, stochastic control, HJB equation, discretisation of control problems.

(23 pages, 2002)

32. M. Krekel

Optimal portfolios with a loan dependent credit spread

If an investor borrows money he generally has to pay higher interest rates than he would have received, if he had put his funds on a savings account. The classical model of continuous time portfolio optimisation ignores this effect. Since there is obviously a connection between the default probability and the total percentage of wealth, which the investor is in debt, we study portfolio optimisation with a control dependent interest rate. Assuming a logarithmic and a power utility function, respectively, we prove explicit formulae of the optimal control.

Keywords: Portfolio optimisation, stochastic control, HJB equation, credit spread, log utility, power utility, non-linear wealth dynamics

(25 pages, 2002)

33. J. Ohser, W. Nagel, K. Schladitz

The Euler number of discretized sets - on the choice of adjacency in homogeneous lattices

Two approaches for determining the Euler-Poincaré characteristic of a set observed on lattice points are considered in the context of image analysis { the integral geometric and the polyhedral approach. Information about the set is assumed to be available on lattice points only. In order to retain properties of the Euler number and to provide a good approximation of the true Euler number of the original set in the Euclidean space, the appropriate choice of adjacency in the lattice for the set and its background is crucial. Adjacencies are defined using tessellations of the whole space into polyhedrons. In R 3 , two new 14 adjacencies are introduced additionally to the well known 6 and 26 adjacencies. For the Euler number of a set and its complement, a consistency relation holds. Each of the pairs of adjacencies (14:1; 14:1), (14:2; 14:2), (6; 26), and (26; 6) is shown to be a pair of complementary adjacencies with respect to this relation. That is, the approximations of the Euler numbers are consistent if the set and its background (complement) are equipped with this pair of adjacencies. Furthermore, sufficient conditions for the correctness of the approximations of the Euler number are given. The analysis of selected microstructures and a simulation study illustrate how the estimated Euler number depends on the chosen adjacency. It also shows that there is not a uniquely best pair of adjacencies with respect to the estimation of the Euler number of a set in Euclidean space.

Keywords: image analysis, Euler number, neighborhod relationships, cuboidal lattice

(32 pages, 2002)

34. I. Ginzburg, K. Steiner

Lattice Boltzmann Model for Free-Surface flow and Its Application to Filling Process in Casting

A generalized lattice Boltzmann model to simulate free-surface is constructed in both two and three dimensions. The proposed model satisfies the interfacial boundary conditions accurately. A distinctive feature of the model is that the collision processes is carried out only on the points occupied partially or fully by the fluid. To maintain a sharp interfacial front, the method includes an anti-diffusion algorithm. The unknown distribution functions at the interfacial region are constructed according to the first order Chapman-Enskog analysis. The interfacial boundary conditions are satis-

fied exactly by the coefficients in the Chapman-Enskog expansion. The distribution functions are naturally expressed in the local interfacial coordinates. The macroscopic quantities at the interface are extracted from the least-square solutions of a locally linearized system obtained from the known distribution functions. The proposed method does not require any geometric front construction and is robust for any interfacial topology. Simulation results of realistic filling process are presented: rectangular cavity in two dimensions and Hammer box, Campbell box, Sheffield box, and Motorblock in three dimensions. To enhance the stability at high Reynolds numbers, various upwind-type schemes are developed. Free-slip and no-slip boundary conditions are also discussed.
Keywords: Lattice Boltzmann models; free-surface phenomena; interface boundary conditions; filling processes; injection molding; volume of fluid method; interface boundary conditions; advection-schemes; upwind-schemes
(54 pages, 2002)

35. M. Günther, A. Klar, T. Materne, R. Wegener

Multivalued fundamental diagrams and stop and go waves for continuum traffic equations

In the present paper a kinetic model for vehicular traffic leading to multivalued fundamental diagrams is developed and investigated in detail. For this model phase transitions can appear depending on the local density and velocity of the flow. A derivation of associated macroscopic traffic equations from the kinetic equation is given. Moreover, numerical experiments show the appearance of stop and go waves for highway traffic with a bottleneck.
Keywords: traffic flow, macroscopic equations, kinetic derivation, multivalued fundamental diagram, stop and go waves, phase transitions
(25 pages, 2002)

36. S. Feldmann, P. Lang, D. Prätzel-Wolters

Parameter influence on the zeros of network determinants

To a network N(q) with determinant D(s;q) depending on a parameter vector $q \hat{I} R^r$ via identification of some of its vertices, a network N^ (q) is assigned. The paper deals with procedures to find N^ (q), such that its determinant D^ (s;q) admits a factorization in the determinants of appropriate subnetworks, and with the estimation of the deviation of the zeros of D^ from the zeros of D. To solve the estimation problem state space methods are applied.
Keywords: Networks, Equicofactor matrix polynomials, Realization theory, Matrix perturbation theory
(30 pages, 2002)

37. K. Koch, J. Ohser, K. Schladitz

Spectral theory for random closed sets and estimating the covariance via frequency space

A spectral theory for stationary random closed sets is developed and provided with a sound mathematical basis. Definition and proof of existence of the Bartlett spectrum of a stationary random closed set as well as the proof of a Wiener-Khintchine theorem for the power spectrum are used to two ends: First, well known second order characteristics like the covariance can be estimated faster than usual via frequency space. Second, the Bartlett spectrum and the power spectrum can be used as second order characteristics in frequency space. Examples show, that in some cases information about the random closed set is easier to obtain from these characteristics in frequency space than from their real world counterparts.
Keywords: Random set, Bartlett spectrum, fast Fourier transform, power spectrum
(28 pages, 2002)

38. D. d'Humières, I. Ginzburg

Multi-reflection boundary conditions for lattice Boltzmann models

We present a unified approach of several boundary conditions for lattice Boltzmann models. Its general framework is a generalization of previously introduced schemes such as the bounce-back rule, linear or quadratic interpolations, etc. The objectives are two fold: first to give theoretical tools to study the existing boundary conditions and their corresponding accuracy; secondly to design formally third- order accurate boundary conditions for general flows. Using these boundary conditions, Couette and Poiseuille flows are exact solution of the lattice Boltzmann models for a Reynolds number $Re = 0$ (Stokes limit).
Numerical comparisons are given for Stokes flows in periodic arrays of spheres and cylinders, linear periodic array of cylinders between moving plates and for Navier-Stokes flows in periodic arrays of cylinders for $Re < 200$. These results show a significant improvement of the overall accuracy when using the linear interpolations instead of the bounce-back reflection (up to an order of magnitude on the hydrodynamics fields). Further improvement is achieved with the new multi-reflection boundary conditions, reaching a level of accuracy close to the quasi-analytical reference solutions, even for rather modest grid resolutions and few points in the narrowest channels. More important, the pressure and velocity fields in the vicinity of the obstacles are much smoother with multi-reflection than with the other boundary conditions.
Finally the good stability of these schemes is highlighted by some simulations of moving obstacles: a cylinder between flat walls and a sphere in a cylinder.
Keywords: lattice Boltzmann equation, boudary condistions, bounce-back rule, Navier-Stokes equation
(72 pages, 2002)

39. R. Korn

Elementare Finanzmathematik

Im Rahmen dieser Arbeit soll eine elementar gehaltene Einführung in die Aufgabenstellungen und Prinzipien der modernen Finanzmathematik gegeben werden. Insbesondere werden die Grundlagen der Modellierung von Aktienkursen, der Bewertung von Optionen und der Portfolio-Optimierung vorgestellt. Natürlich können die verwendeten Methoden und die entwickelte Theorie nicht in voller Allgemeinheit für den Schulunterricht verwendet werden, doch sollen einzelne Prinzipien so heraus gearbeitet werden, dass sie auch an einfachen Beispielen verstanden werden können.
Keywords: Finanzmathematik, Aktien, Optionen, Portfolio-Optimierung, Börse, Lehrerweiterbildung, Mathematikunterricht
(98 pages, 2002)

40. J. Kallrath, M. C. Müller, S. Nickel

Batch Presorting Problems: Models and Complexity Results

In this paper we consider short term storage systems. We analyze presorting strategies to improve the effiency of these storage systems. The presorting task is called Batch PreSorting Problem (BPSP). The BPSP is a variation of an assigment problem, i. e., it has an assigment problem kernel and some additional constraints. We present different types of these presorting problems, introduce mathematical programming formulations and prove the NP-completeness for one type of the BPSP. Experiments are carried out in order to compare the different model formulations and to investigate the behavior of these models.
Keywords: Complexity theory, Integer programming, Assigment, Logistics
(19 pages, 2002)

41. J. Linn

On the frame-invariant description of the phase space of the Folgar-Tucker equation

The Folgar-Tucker equation is used in flow simulations of fiber suspensions to predict fiber orientation depending on the local flow. In this paper, a complete, frame-invariant description of the phase space of this differential equation is presented for the first time.
Key words: fiber orientation, Folgar-Tucker equation, injection molding
(5 pages, 2003)

42. T. Hanne, S. Nickel

A Multi-Objective Evolutionary Algorithm for Scheduling and Inspection Planning in Software Development Projects

In this article, we consider the problem of planning inspections and other tasks within a software development (SD) project with respect to the objectives quality (no. of defects), project duration, and costs. Based on a discrete-event simulation model of SD processes comprising the phases coding, inspection, test, and rework, we present a simplified formulation of the problem as a multiobjective optimization problem. For solving the problem (i. e. finding an approximation of the efficient set) we develop a multiobjective evolutionary algorithm. Details of the algorithm are discussed as well as results of its application to sample problems.
Key words: multiple objective programming, project management and scheduling, software development, evolutionary algorithms, efficient set
(29 pages, 2003)

43. T. Bortfeld , K.-H. Küfer, M. Monz, A. Scherrer, C. Thieke, H. Trinkaus

Intensity-Modulated Radiotherapy - A Large Scale Multi-Criteria Programming Problem -

Radiation therapy planning is always a tight rope walk between dangerous insufficient dose in the target volume and life threatening overdosing of organs at risk. Finding ideal balances between these inherently contradictory goals challenges dosimetrists and physicians in their daily practice. Today's planning systems are typically based on a single evaluation function that measures the quality of a radiation treatment plan. Unfortunately, such a one dimensional approach can-

not satisfactorily map the different backgrounds of physicians and the patient dependent necessities. So, too often a time consuming iteration process between evaluation of dose distribution and redefinition of the evaluation function is needed.
In this paper we propose a generic multi-criteria approach based on Pareto's solution concept. For each entity of interest - target volume or organ at risk a structure dependent evaluation function is defined measuring deviations from ideal doses that are calculated from statistical functions. A reasonable bunch of clinically meaningful Pareto optimal solutions are stored in a data base, which can be interactively searched by physicians. The system guarantees dynamical planning as well as the discussion of tradeoffs between different entities.
Mathematically, we model the upcoming inverse problem as a multi-criteria linear programming problem. Because of the large scale nature of the problem it is not possible to solve the problem in a 3D-setting without adaptive reduction by appropriate approximation schemes.
Our approach is twofold: First, the discretization of the continuous problem is based on an adaptive hierarchical clustering process which is used for a local refinement of constraints during the optimization procedure. Second, the set of Pareto optimal solutions is approximated by an adaptive grid of representatives that are found by a hybrid process of calculating extreme compromises and interpolation methods.
Keywords: multiple criteria optimization, representative systems of Pareto solutions, adaptive triangulation, clustering and disaggregation techniques, visualization of Pareto solutions, medical physics, external beam radiotherapy planning, intensity modulated radiotherapy
(31 pages, 2003)

44. T. Halfmann, T. Wichmann

Overview of Symbolic Methods in Industrial Analog Circuit Design

Industrial analog circuits are usually designed using numerical simulation tools. To obtain a deeper circuit understanding, symbolic analysis techniques can additionally be applied. Approximation methods which reduce the complexity of symbolic expressions are needed in order to handle industrial-sized problems. This paper will give an overview to the field of symbolic analog circuit analysis. Starting with a motivation, the state-of-the-art simplification algorithms for linear as well as for nonlinear circuits are presented. The basic ideas behind the different techniques are described, whereas the technical details can be found in the cited references. Finally, the application of linear and nonlinear symbolic analysis will be shown on two example circuits.
Keywords: CAD, automated analog circuit design, symbolic analysis, computer algebra, behavioral modeling, system simulation, circuit sizing, macro modeling, differential-algebraic equations, index
(17 pages, 2003)

45. S. E. Mikhailov, J. Orlik

Asymptotic Homogenisation in Strength and Fatigue Durability Analysis of Composites

Asymptotic homogenisation technique and two-scale convergence is used for analysis of macro-strength and fatigue durability of composites with a periodic structure under cyclic loading. The linear damage accumulation rule is employed in the phenomenological micro-durability conditions (for each component of the composite) under varying cyclic loading. Both local and non-local strength and durability conditions are analysed. The strong convergence of the strength and fatigue damage measure as the structure period tends to zero is proved and their limiting values are estimated.
Keywords: multiscale structures, asymptotic homogenization, strength, fatigue, singularity, non-local conditions
(14 pages, 2003)

46. P. Domínguez-Marín, P. Hansen, N. Mladenović, S. Nickel

Heuristic Procedures for Solving the Discrete Ordered Median Problem

We present two heuristic methods for solving the Discrete Ordered Median Problem (DOMP), for which no such approaches have been developed so far. The DOMP generalizes classical discrete facility location problems, such as the p-median, p-center and Uncapacitated Facility Location problems. The first procedure proposed in this paper is based on a genetic algorithm developed by Moreno Vega [MV96] for p-median and p-center problems. Additionally, a second heuristic approach based on the Variable Neighborhood Search metaheuristic (VNS) proposed by Hansen & Mladenovic [HM97] for the p-median problem is described. An extensive numerical study is presented to show the efficiency of both heuristics and compare them.
Keywords: genetic algorithms, variable neighborhood search, discrete facility location
(31 pages, 2003)

47. N. Boland, P. Domínguez-Marín, S. Nickel, J. Puerto

Exact Procedures for Solving the Discrete Ordered Median Problem

The Discrete Ordered Median Problem (DOMP) generalizes classical discrete location problems, such as the N-median, N-center and Uncapacitated Facility Location problems. It was introduced by Nickel [16], who formulated it as both a nonlinear and a linear integer program. We propose an alternative integer linear programming formulation for the DOMP, discuss relationships between both integer linear programming formulations, and show how properties of optimal solutions can be used to strengthen these formulations. Moreover, we present a specific branch and bound procedure to solve the DOMP more efficiently. We test the integer linear programming formulations and this branch and bound method computationally on randomly generated test problems.
Keywords: discrete location, Integer programming
(41 pages, 2003)

48. S. Feldmann, P. Lang

Padé-like reduction of stable discrete linear systems preserving their stability

A new stability preserving model reduction algorithm for discrete linear SISO-systems based on their impulse response is proposed. Similar to the Padé approximation, an equation system for the Markov parameters involving the Hankel matrix is considered, that here however is chosen to be of very high dimension. Although this equation system therefore in general cannot be solved exactly, it is proved that the approximate solution, computed via the Moore-Penrose inverse, gives rise to a stability preserving reduction scheme, a property that cannot be guaranteed for the Padé approach. Furthermore, the proposed algorithm is compared to another stability preserving reduction approach, namely the balanced truncation method, showing comparable performance of the reduced systems. The balanced truncation method however starts from a state space description of the systems and in general is expected to be more computational demanding.
Keywords: Discrete linear systems, model reduction, stability, Hankel matrix, Stein equation
(16 pages, 2003)

49. J. Kallrath, S. Nickel

A Polynomial Case of the Batch Presorting Problem

This paper presents new theoretical results for a special case of the batch presorting problem (BPSP). We will show tht this case can be solved in polynomial time. Offline and online algorithms are presented for solving the BPSP. Competetive analysis is used for comparing the algorithms.
Keywords: batch presorting problem, online optimization, competetive analysis, polynomial algorithms, logistics
(17 pages, 2003)

50. T. Hanne, H. L. Trinkaus

knowCube for MCDM – Visual and Interactive Support for Multicriteria Decision Making

In this paper, we present a novel multicriteria decision support system (MCDSS), called knowCube, consisting of components for knowledge organization, generation, and navigation. Knowledge organization rests upon a database for managing qualitative and quantitative criteria, together with add-on information. Knowledge generation serves filling the database via e. g. identification, optimization, classification or simulation. For "finding needles in haycocks", the knowledge navigation component supports graphical database retrieval and interactive, goal-oriented problem solving. Navigation "helpers" are, for instance, cascading criteria aggregations, modifiable metrics, ergonomic interfaces, and customizable visualizations. Examples from real-life projects, e.g. in industrial engineering and in the life sciences, illustrate the application of our MCDSS.
Key words: Multicriteria decision making, knowledge management, decision support systems, visual interfaces, interactive navigation, real-life applications.
(26 pages, 2003)

51. O. Iliev, V. Laptev

On Numerical Simulation of Flow Through Oil Filters

This paper concerns numerical simulation of flow through oil filters. Oil filters consist of filter housing (filter box), and a porous filtering medium, which completely separates the inlet from the outlet. We discuss mathematical models, describing coupled flows in the pure liquid subregions and in the porous filter media, as well as interface conditions between them. Further, we reformulate the problem in fictitious regions method manner, and discuss peculiarities of the numerical algorithm in solving the coupled system. Next, we show numerical results, validating the model and the

algorithm. Finally, we present results from simulation of 3-D oil flow through a real car filter.
Keywords: oil filters, coupled flow in plain and porous media, Navier-Stokes, Brinkman, numerical simulation
(8 pages, 2003)

52. W. Dörfler, O. Iliev, D. Stoyanov, D. Vassileva

On a Multigrid Adaptive Refinement Solver for Saturated Non-Newtonian Flow in Porous Media

A multigrid adaptive refinement algorithm for non-Newtonian flow in porous media is presented. The saturated flow of a non-Newtonian fluid is described by the continuity equation and the generalized Darcy law. The resulting second order nonlinear elliptic equation is discretized by a finite volume method on a cell-centered grid. A nonlinear full-multigrid, full-approximation-storage algorithm is implemented. As a smoother, a single grid solver based on Picard linearization and Gauss-Seidel relaxation is used. Further, a local refinement multigrid algorithm on a composite grid is developed. A residual based error indicator is used in the adaptive refinement criterion. A special implementation approach is used, which allows us to perform unstructured local refinement in conjunction with the finite volume discretization. Several results from numerical experiments are presented in order to examine the performance of the solver.
Keywords: Nonlinear multigrid, adaptive refinement, non-Newtonian flow in porous media
(17 pages, 2003)

53. S. Kruse

On the Pricing of Forward Starting Options under Stochastic Volatility

We consider the problem of pricing European forward starting options in the presence of stochastic volatility. By performing a change of measure using the asset price at the time of strike determination as a numeraire, we derive a closed-form solution based on Heston's model of stochastic volatility.
Keywords: Option pricing, forward starting options, Heston model, stochastic volatility, cliquet options
(11 pages, 2003)

54. O. Iliev, D. Stoyanov

Multigrid – adaptive local refinement solver for incompressible flows

A non-linear multigrid solver for incompressible Navier-Stokes equations, exploiting finite volume discretization of the equations, is extended by adaptive local refinement. The multigrid is the outer iterative cycle, while the SIMPLE algorithm is used as a smoothing procedure. Error indicators are used to define the refinement subdomain. A special implementation approach is used, which allows to perform unstructured local refinement in conjunction with the finite volume discretization. The multigrid - adaptive local refinement algorithm is tested on 2D Poisson equation and further is applied to a lid-driven flows in a cavity (2D and 3D case), comparing the results with bench-mark data. The software design principles of the solver are also discussed.
Keywords: Navier-Stokes equations, incompressible flow, projection-type splitting, SIMPLE, multigrid methods, adaptive local refinement, lid-driven flow in a cavity
(37 pages, 2003)

55. V. Starikovicius

The multiphase flow and heat transfer in porous media

In first part of this work, summaries of traditional Multiphase Flow Model and more recent Multiphase Mixture Model are presented. Attention is being paid to attempts include various heterogeneous aspects into models. In second part, MMM based differential model for two-phase immiscible flow in porous media is considered. A numerical scheme based on the sequential solution procedure and control volume based finite difference schemes for the pressure and saturation-conservation equations is developed. A computer simulator is built, which exploits object-oriented programming techniques. Numerical result for several test problems are reported.
Keywords: Two-phase flow in porous media, various formulations, global pressure, multiphase mixture model, numerical simulation
(30 pages, 2003)

Status quo: July 2003