Fraunhofer Institut
Techno- und
Wirtschaftsmathematik

R. Ciegis, O. Iliev, Z. Lakdawala

On parallel numerical algorithms for simulating industrial filtration problems

Berichte des Fraunhofer ITWM, Nr. 114 (2007)

ISSN 1434-9973

Bericht 114 (2007)

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

Fraunhofer-Institut für Techno- und
Wirtschaftsmathematik ITWM
Fraunhofer-Platz 1

67663 Kaiserslautern
Germany

Telefon: +49 (0) 6 31/3 16 00-0
Telefax: +49 (0) 6 31/3 16 00-10 99
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.

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

Berichterstattung heißt hier Dokumentation des Transfers aktueller Ergebnisse aus mathematischer Forschungs- und Entwicklungsarbeit in industrielle Anwendungen und Softwareprodukte – und umgekehrt, denn Probleme der Praxis generieren neue interessante mathematische Fragestellungen.

Prof. Dr. Dieter Prätzel-Wolters
Institutsleiter

Kaiserslautern, im Juni 2001

On Parallel Numerical Algorithms for Simulating Industrial Filtration Problems

R. Čiegis,
Vilnius Gediminas Technical University
Saulėtekio al. 11, LT-10223, Vilnius, Lithuania
e-mail: rc@fm.vtu.lt

O. Iliev, Z. Lakdawala
Fraunhofer ITWM
Fraunhofer-Platz 1, D-67663 Kaiserslautern, Germany
e-mail: {iliev, lakdawala}@itwm.fhg.de

April 23, 2007

Abstract

The performance of oil filters used in the automotive industry can be significantly improved, especially when computer simulation is an essential component of the design process. In this paper, we consider parallel numerical algorithms for solving mathematical models describing the process of filtration, filtering out solid particles from liquid oil. The Navier-Stokes-Brinkmann system of equations is used to describe the laminar flow of incompressible isothermal oil. The space discretization in the complicated filter geometry is based on the finite-volume method. Special care is taken for an accurate approximation of velocity and pressure on the interface between the fluid and the porous media. The time discretization used here is a proper modification of the fractional time step discretization (cf. Chorin scheme) of the Navier-Stokes equations, where the Brinkmann term is considered at both, prediction and correction substeps.

A data decomposition method is used to develop a parallel algorithm, where the domain is distributed among processors by using a structured reference grid. The MPI library is used to implement the data communication part of the algorithm. A theoretical model is proposed for the estimation of the complexity of the given parallel algorithm and a scalability analysis is done on the basis of this model. Results of computational experiments are presented, and the accuracy

and efficiency of the parallel algorithm is tested on real industrial geometries.

1 Introduction

The numerical solution of nonlinear PDEs, such as porous media models or Navier-Stokes equations, require very large computational efforts. A significant step in reducing the CPU time and/or increasing the accuracy of the simulations is the usage of parallel computers and clusters of workstations. The power of modern personal computers is increasing constantly, but not enough to fulfill all scientific and engineering computational demands. In such cases, parallel computing may be the answer. Parallel computing not only gives access to increasing computational resources but it also becomes economically feasible. This is mainly because clusters of workstations can be used as local dedicated computational nodes or as a parallel computer, according to momental needs of a department. A good review on the state of the art in numerical solution of PDEs on parallel computers is given in [7]. This book surveys the major topics that are essential to high-performance simulation on parallel computers, including programming models, load balancing, mesh generation, efficient numerical solvers, and scientific software.

Filtering out solid particles from liquid oil is very essential for automotive engines (as well as for many other applications). An oil filter can be described shortly as a filter box (which could be of complicated shape) with inlet/s for dirty oil and outlet/s for filtrated oil. The inlet/s and outlet/s are separated by a filtering medium, which is usually a single layer or a multilayer porous media. Optimal shape design for the filter housing, achieving optimal pressure drop - flow rate ratio, etc., require detailed knowledge about the flow field through the filter. Accurate information about the velocity and pressure distributions can be obtained from a 3-D computer simulation of the fully coupled flow: flow in the area between the inlet/s and the filtering medium, flow within filtering medium, and flow between the filtering medium and the outlet/s. A general purpose commercial CFD software might be inefficient and/or inaccurate in computing such particular flows. An efficient numerical algorithm for such problems is developed. The algorithm is further implemented to meet the needs of an existing manufacturing oil filter company, namely IBS Filtran. Intensive collaboration with the company achieved success with the release of a specialized software, namely SuFiS (Suction Filter Simulation). The single grid version of the algorithm is shortly presented in [19, 20] (for some details, see [27]).

The top and bottom pans of a typical filter housing are shown in Fig.1, and a sketch of a cross-section of a filter assembly is shown in Fig.2. Several challenging problems have to be solved to support the design of oil filters: detailed simulation of coupled flows through filters; modeling and simulation of capturing of dirt particles by the filtering media; interaction of the flow and the deformable filtering medium; optimal shape design and so on. Here we discuss the first of these problems, i.e. the coupled flow simulations. More precisely, we discuss the development of efficient numerical algorithms and software for the simulation of 3-D flow through oil filters. We also pay special attention to the usage of correct parameters and to the validation of the software. Note that some details concerning modelling and simulation of capturing of the particles by the filtering medium, as well as of related filtration processes, can be found in [16].

Figure 1: Top pan (left) and bottom pan (right) of a filter housing;

Figure 2: A schematic drawing of a vertical cross-section of the filter

This paper aims to discuss parallelization of the existing industrial sequential code SuFiS, which is extensively used for the simulation of fluid flows in industrial filters. The domain or data parallelization paradigm is used to build a parallel algorithm [25]. The MPI library is used to implement the data communication among processors.

The paper is organized as follows. In Section 2, we first formulate the problem. The following two subsections describe the fractional time step discretization, and the Finite Volume (FV) discretization method used for solving the flow problem, respectively. A parallel algorithm is described

in Section 3, which is based on the parallel domain (data) decomposition method. The systems of linear equations are solved by a parallel version of the preconditioned BiConjugate Gradient Stabilized algorithm. A theoretical model, which estimates the complexity of the parallel SuFiS algorithm is proposed. The results of computational experiments corresponding to the SP5 computer and a cluster of workstations are presented and the efficiency of some popular parallel preconditioners is investigated. During computations, the diagonal and the incomplete LU (ILU) factorization preconditioners are considered. A parallel version of ILU is obtained by doing the factorization of a local part of the matrix at each processor. It is well known that such strategy reduces the convergence rate of the iterative algorithm, however, the parallelism of the obtained preconditioner is the same as the one obtained for the diagonal one. In Section 4, the computational results of experiments are presented, where some industrial filters are simulated and flows in such filters are investigated. Some final conclusions are given in Section 5.

2 Governing equations and space and time discretization

2.1 Governing equations

The Brinkmann model (see, e.g. [6, 24]) describing the flow in porous media, Ω_p, and the Navier-Stokes equations (see, e.g. [13]) describing the flow in the pure fluid region, Ω_f, together with the interface conditions for the continuity of the velocity and the continuity of the normal component of the stress tensor, are reformulated such that a single system of partial differential equations governs the flow in the pure liquid and in the porous media. This is done using the fictitious regions method manner. Note that the fictitious region method allows the use of one system of equations in order to treat the fluid, porous and solid regions simultaneously (see, e.g. [1, 30, 31]). The coefficients of the equations vary in a way such that the single system is reduced to the Navier-Stokes equations in the liquid zone, and to the Brinkmann–like model in the porous media. This approach is relevant when the interface conditions between the plain (fluid) and porous media are chosen to be the continuity of the velocity and the continuity of the normal component of the stress tensor. For a theoretical justification of this approach, for the case of Stokes flow, see [1, 30] and the references therein.

The Navier-Stokes-Brinkmann system of equations describing laminar, incompressible and isothermal flow in the whole domain reads:

$$\begin{cases} \underbrace{\dfrac{\partial \rho \vec{u}}{\partial t} - \nabla \cdot \left(\tilde{\mu} \nabla \vec{u}\right) + (\rho \vec{u}, \nabla)\vec{u}}_{\text{Navier-Stokes}} + \overbrace{\tilde{\mu}\tilde{\mathbf{K}}^{-1}\vec{u} + \underbrace{\nabla p = \tilde{\vec{f}}}}^{\text{Darcy law}}, \\ \nabla \cdot \vec{u} = 0. \end{cases} \tag{1}$$

In this case, the tilde-quantities are defined as follows:

$$\tilde{\mu} = \begin{cases} \mu & in\ \Omega_f, \\ \mu_{eff} & in\ \Omega_p, \end{cases} \quad \tilde{\vec{f}} = \begin{cases} \vec{f}_{NS} & in\ \Omega_f, \\ \vec{f}_B & in\ \Omega_p, \end{cases} \quad \tilde{\mathbf{K}}^{-1} = \begin{cases} 0 & in\ \Omega_f, \\ \mathbf{K}^{-1} & in\ \Omega_p. \end{cases}$$

Here $\vec{u}$ and p stand for the velocity vector and the pressure, respectively, and $\rho, \mu, \mathbf{K}$ denote the density, the viscosity, and the permeability tensor of the porous medium, respectively.

No slip boundary conditions are prescribed on the solid wall, prescribed flow rate (i.e., prescribed velocity) are the boundary conditions at the inflow, and soft boundary conditions are prescribed at the outflow.

2.2 Time Discretization

The choice of the time discretization influences the accuracy of the numerical solution and the stability of the algorithm (e.g., the restrictions on the time step). In this paper, we are interested in the steady state solution only, when the steady state solution is a limit of the unsteady one for $t \to \infty$. Thus stability and fast convergence are the criteria influencing the choice of the time discretization here. Studies of the unsteady regimes will be the subject of another paper. Before the time discretizations are presented, some notations are introduced. The operators corresponding to the discretized convective and diffusive terms in the momentum equations are denoted by $\mathrm{C}(\vec{u})\vec{u}$ and $D\vec{u}$, respectively. The particular form of these operators depend on the space discretization, and will be discussed below. Further, the discretization of the gradient is denoted by G, and G^T denotes the discretization of the divergence operator. Finally, $\mathrm{B}\vec{u}$ denoted the operator corresponding to Darcy term, namely $\mu\tilde{\mathbf{K}}^{-1}\vec{u}$, in the momentum equations. In some cases, the same notation is used for the continuous and discrete functions, when this does not cause misunderstanding. Below, the superscript ^{n} is used to denote the values at the old time level, and $^{n+1}$ or NO superscript to

denote the values at the new time level. Notation τ stands for the time step, $\tau = t^{n+1} - t^n$. With these notations, the following fractional time step discretization scheme can be written:

$$\begin{aligned}
\left(\rho\vec{u}^{n+\frac{1}{2}} - \rho\vec{u}^{n}\right) + \tau\left(C(\vec{u}^{n}) - D + B\right)\ \vec{u}^{n+\frac{1}{2}} &= \tau\ G\ p^{n}, \\
\left(\rho\vec{u^{n+1}} - \rho\vec{u}^{n+\frac{1}{2}}\right) + \tau\left(B\vec{u}^{n+1} - B\vec{u}^{n+\frac{1}{2}}\right) &= \tau\left(G\ p^{n+1} - G\ p^{n}\right), \qquad (2) \\
G^T\rho\vec{u^{n+1}} &= 0.
\end{aligned}$$

This algorithm for the Navier-Stokes-Brinkmann equations can be viewed as a modification of the well-known Chorin method for Navier-Stokes equations. The sum of the first and the second equations above gives an implicit discretization of the momentum equations. The first equation is solved with respect to the velocities, using the old value of the pressure gradient, thus obtaining a prediction for the velocity. To solve the second equation with respect to pressure correction, one takes the divergence from it and uses the continuity equation. The result is a Poisson-type equation for the pressure correction. There exists extensive mathematical literature, concerning first and second order fractional time step discretizations, incremental and non-incremental form of equations, stability, splitting of the boundary conditions, etc. Some discussions on this topic, as well as further references, can be found in [14, 8].

As mentioned above, the Darcy term in the Navier-Stokes-Brinkmann equations needs special treatment. In equation (2), this term is accounted for at both time substeps. Note that the pressure correction equation should be carefully derived in this case. A naive application of the Chorin method would give:

$$G^T\left(\rho\vec{u}^{n+1} - \rho\vec{u}^{n+\frac{1}{2}}\right) + G^T\tau\left(B\vec{u}^{n+1} - B\vec{u}^{n+\frac{1}{2}}\right) = G^T\tau\left(G\ p^{n+1} - G\ p^{n}\right). \qquad (3)$$

Denoting q for the pressure correction, where $q = p^{n+1} - p^n$, one can rewrite the above equation as

$$G^T\tau Gq = G^T\left(\rho\vec{u}^{n+1} - \rho\vec{u}^{n+\frac{1}{2}}\right) + G^T\tau\left(B\vec{u}^{n+1} - B\vec{u}^{n+\frac{1}{2}}\right). \qquad (4)$$

Furthermore, using the continuity equation $G^T\rho u^{n+1} = 0$, and assuming that $G^T B u^{n+1} \approx 0$, the above equation reduces to

$$G^T\tau Gq = -G^T\left(\rho\vec{u}^{n+\frac{1}{2}} + \tau B\vec{u}^{n+\frac{1}{2}}\right). \qquad (5)$$

A drawback of this pressure correction equation is that its operator does not "see" the porous media, and that the continuity equation is approximately satisfied in the porous media. For a constant time step, its operator is equivalent to Poisson equation with constant coefficients.

In order to achieve a better convergence, let us consider another approach for forming a pressure correction equation. Denote I as the identity 3×3 matrix. Rewriting equation (2) gives:

$$\left(I + \frac{\tau}{\rho}B\right)\rho\vec{u}^{n+1} - \left(I + \frac{\tau}{\rho}B\right)\rho\vec{u}^{n+\frac{1}{2}} = \tau\left(G\ p^{n+1} - G\ p^{n}\right). \tag{6}$$

Keeping in mind that $(I + \frac{\tau}{\rho}B)^{-1}$ always exists because B is positive definite, the above equation can be transformed to

$$\left(\rho\vec{u}^{n+1} - \rho\vec{u}^{n+\frac{1}{2}}\right) = (I + \frac{\tau}{\rho}B)^{-1}\tau\left(G\ p^{n+1} - G\ p^{n}\right). \tag{7}$$

Now, applying the divergence operator, G^T, to this equation, and using the continuity equation, the following pressure correction equation is obtained:

$$G^T(I + \frac{\tau}{\rho}B)^{-1}\tau G\ q = -G^T\rho\vec{u}^{n+\frac{1}{2}}. \tag{8}$$

It is easy to see that the derived pressure correction equation in the pure fluid region, where $B = 0$, reduces to the standard Chorin scheme. The same equation for pure fluids is obtained within the SIMPLEC approach (see [13]), or within the Schur complement approach (see [32]). It is more important to see that this equation has the form of the well-known Darcy equation in the porous medium (note that in the porous medium $I \ll B$). Thus, the equation describes both, the pure fluid zones and the porous zones equally well.

After the pressure correction equation is solved, the pressure is updated, $p^{n+1} = p^n + q$, and the new velocity is calculated based on equation (7):

$$\rho\vec{u}^{n+1} = \rho\vec{u}^{n+\frac{1}{2}} + (I + \frac{\tau}{\rho}B)^{-1}\tau Gq \tag{9}$$

2.3 Grid and Finite Volume Discretization in space

The geometrical information about the computational domain is usually provided in a CAD format, for example, the *stl* format. A pre-processor

based on Level Set Method (see [15]), is used in order to process the given CAD data for attaining the assembly of a filter housing. The output of the pre-processor is a computational domain (i.e., the internal volume of the filter housing), along with a generated grid. An example of a computational domain can be seen in Figure 3.

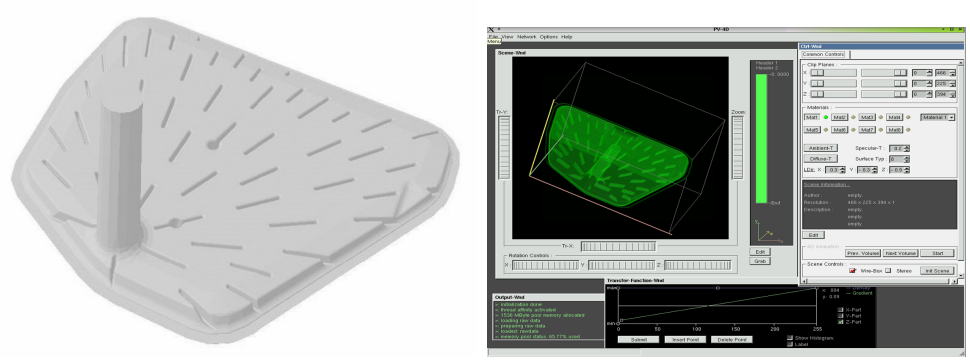

Figure 3: Computational domain of a filter housing (left) and a snapshot of the visualization tool (right)

The governing equations are discretized by the Finite Volume Method (see [13]) on the generated Cartesian grid. Cell-centered grid with collocated arrangement of the velocity and pressure is used. The Rhie-Chow interpolation (see [12]) is used to avoid the spiral oscillations, which could appear due to the collocated arrangement of the unknowns. Upwind, central differencing, and deferred correction schemes can be used for the convective term. In general, the discretization of the convective and the diffusive (viscous) terms in the pure fluid region is close or identical to the one described in [12, 13], and therefore it will not be described here (some details can be found in our earlier papers, for example, [20, 21, 22]. Special attention is paid to the discretizations near the interfaces between the fluid and the porous medium. Conservativity of the discretization is achieved here by choosing finite volume method as a discretization technique. To get an accurate approximation for the velocity and for the pressure on the interface, a special modification of the discretizations near the interface is used. First of all, it should be noted that the pressure gradient in the momentum equation is discretized in each cell separately. To do this, the pressure values from the cell centers are interpolated to the cell faces. In the pure fluid region, this is done by linear interpolation where the pressure gradient is discretized via central differences. The more crucial part is the interpolation of pressure

on the interface between the pure fluid and the porous regions. Problem-dependent interpolation employed, using the operator from (8) together with the mass conservation assumption (i.e., continuity of the normal component of the flux across the interface). For a detailed illustration of this approach, suppose that p_L is a pressure value in a pure fluid cell located left from the interface, and p_R is a pressure value in a porous cell located right from the interface. The pressure at the interface is assumed to be continuous, and the value there, p_e, in the case of isotropic media (i.e. scalar permeability) is calculated from (8) as follows:

$$p_e = \frac{(1 + \frac{\tau}{\rho} b)_L^{-1} p_L + (1 + \frac{\tau}{\rho} b)_R^{-1} p_R}{(1 + \frac{\tau}{\rho} b)_L^{-1} + (1 + \frac{\tau}{\rho} b)_R^{-1}}, \tag{10}$$

where b is the component of B in the isotropic case.

2.4 Series of computations with different velocities and viscosities

In practice, a series of computations with different velocities and different viscosities need to be performed for a fixed geometry. Particularly, in the case of oil filter simulations, the performance of a filter is usually evaluated at different flow rates and different temperatures (the last results in different viscosities and different densities). Also, recall that each single computation is performed for incompressible nonisothermal fluid. In order to reduce the computational efforts, a *start from previous* procedure is briefly discussed. This routine takes advantage of the fact that there exists a good initial guess for all simulated cases except for the first one. The cases to be simulated are ordered such that the parameter

$$\gamma_l = \frac{1}{\mu_l Q_l}$$

is increasing. Here Q stands for the prescribed flow rate at the inlet, and l stands for the current number of the set of input parameters. After the first case corresponding to γ_l is computed, the computations corresponding to γ_{l+1} start with reading the l-th steady state solution, and thereby rescaling the pressure in accordance with the formula

$$Gp_{l+1} = \frac{\gamma_l}{\gamma_{l+1}} Gp_l. \tag{11}$$

2.5 Implementational sketch

At the end of this section we summarize the proposed discrete scheme and numerical algorithms. In Figure 4 a short description of the sequential SUFIS algorithm is given.

SUFIS ()
begin
(1) Define computational domain
(2) ok = true; k = 0;
(3) **while** (ok) **do**
(4) k := k+1; $U^k = U^{k-1}$; $P^k = P^{k-1}$;
(5) **for** (i=0; i < MaxNonLinearIter; i++) **do**
(6) Compute velocities from momentum equations
$Q_j U_j^* = F_j^i - \partial_{x_j} P^k, \;\; j = 1, 2, 3;$
(7) Solve equation for the pressure correction
$L_h P_c = R^i;$
(8) Correct the velocities
$U_j^k = U_j^* - \alpha_u (D^{-1} \nabla_h P_c)_j, \;\; j = 1, 2, 3;$
(9) Correct the pressure
$P^k := P^k + \alpha_p P_c;$
end do
(10) **if** (final time step) ok = false;
end do
end SUFIS

Figure 4: Sequential SUFIS algorithm

During each iteration at steps (6) and (7) four systems of linear equations are solved. For many problems this part of the algorithm requires till 90% of total CPU time. We use the BiCGSTAB algorithm which solves the unsymmetric linear system

$$Ax = f$$

by using the preconditioned BiConjugate Gradient Stabilized method [4]. A short description of the sequential BiCGSTAB algorithm is given in Figure 5.

BiCGSTAB (Vec x, Vec f, Matrix A, double ε)
begin

(1) Compute the precondinioner D;
(2) normf = $\|f\|$;
(3) **if** (normf $< \varepsilon$) normf = 1;
(4) $r = f - Ax$;
(5) $\tilde{r} = r$; ok = true; i = 0;
(6) **while** (ok) **do**
(7) i++; $\rho_1 = (\tilde{r}, r)$;
(8) **if** (i == 1) **then**
(9) p = r;
else
(10) $\beta = (\rho_1\alpha)/(\rho_2\omega)$;
(11) $p = r + \beta\,(p - \omega v)$;
end if
(12) $D\hat{p} = p$;
(13) $v = A\hat{p}$;
(14) $\alpha = \rho_1/(\tilde{r}, v)$;
(15) $s = r - \alpha v$;
(16) **if** ($\|s\|$ / normf $< \varepsilon$) **then**
(17) $x+ = \alpha\hat{p}$;
(18) ok = false;
else
(19) $D\hat{s} = s$;
(20) $t = A\hat{s}$;
(21) $\omega = (t, s)/(t, t)$;
(22) $x+ = \alpha\hat{p} + \omega\hat{s}$; $r = s - \omega t$;
(23) $\rho_2 = \rho_1$;
(24) **if** ($\|r\|$ / normf $< \varepsilon$) ok = false;
end if
end do

end BiCGSTAB

Figure 5: Serial BiCGSTAB algorithm

3 Parallel Algorithm

In this section, the main details of the developed parallel algorithm are described. A theoretical model estimating the complexity of the parallel SuFiS algorithm is also proposed.

As stated above, the solver for the linear system of equations typically requires up to 70 to 90 percents of total computation time. Thus there is a possibility to parallelize only the linear system solver by using one of the well-known parallel software packages such as PETSc [3]. Alternatively, in many cases, the main goal of parallel computations is not only to increase the size of simulated problems but also to solve them faster. Therefore, the distribution of the discretization part of the algorithm to scale the problem size according to the increased number of processors becomes a crucial point to consider. Also, when only the linear solver is parallelized, computing the discretization is done only on the master processor using the sequential code while the linear systems of equations are solved in parallel. The drawback of this practice becomes apparent when the additional costs, pertaining to the distribution of the matrix and right-hand side vector among processors and assembling the solution on the master processor, are considered.

3.1 Domain Decomposition

The Navier-Stokes-Brinkmann system of equations (1) is solved in a complicated 3D region. A discrete grid is described as a general non-structured set of finite-volumes. The discretization of the PDEs and assembling of the coefficients of the linear equations is done by using a general technique, which is standard for solvers of flows in porous media and Navier-Stokes equations [5, 26]. The goal of the domain (data) decomposition method is to define a suitable mapping of all finite-volumes V to the set of p processors

$$V = V_1 \cup V_2 \cup \ldots \cup V_p,$$

where V_j defines the elements mapped to j-th processor. The load balancing problem should be solved during the implementation of this step. First, it is aimed that each processor has about the same number of elements, since this number would define the computational complexity for all parts of the SuFiS algorithm. Due to the stencil of discretization, the computational domains of processors can overlap. The information belonging to the overlapped regions should be exchanged among processors. For distributed memory computers, MPI library is used to send explicit messages between processors,

contributing to the additional costs of the parallel algorithm. Thus, a second goal of defining the optimal data mapping is to minimize the overlapping regions.

The p-way graph partitioning problem is NP-complete, i.e. no polynomial time algorithm is likely to be found to solve this problem. Therefore heuristic algorithms are developed to find a good solution in a reasonable time. The multilevel partitioning method is one of the most efficient partitioning methods having a linear time complexity. State of the art implementation of a family of multilevel partitioning methods for partitioning unstructured graphs and hypergraphs is available as a METIS software library [23].

A simpler domain decomposition algorithm has been applied for this particular case. It takes into account that the orthogonal 3D structured grid is used as a reference grid for definition of computational grid. Therefore a standard 3D decomposition of processors $p_1 \times p_2 \times p_3$ can be used. Such a strategy simplifies the implementation of data exchange algorithms, since it is very easy to define the neighbours of each processor and the overlapping elements.

Load balancing. In order to solve the load balancing problem for a given number of processors, all combinations of 3D processors topologies are generated. The topology with the best load balancing and minimizing the number overlapping elements is found.

In a series of computational experiments, the quality of the obtained partitionings were tested and compared with the partitionings computed by METIS. In Table 1, the values of load disbalance parameter d_p and the number of overlapping elements w_p (or edges cutting the partitioned subsets of elements in the case of METIS partitioning) are presented for METIS and orthogonal 3D partitionings. The grid was generated for a real industrial application, the graph of this grid had 596094 nodes and 1507732 edges, the auxiliary structured grid had 5428000 nodes.

It was seen that the simple grid partitioning algorithm gives mappings with good load balancing and the number of overlapped elements is also close to the number of similar elements in partitionings generated by METIS.

Now the costs of data initialization are estimated. The *master* processor reads the information on the grid from a file and broadcasts it to the other processors. The complexity of the global *broadcast* operation depends strongly on the architecture of the parallel computer, see [10, 17, 18]. The cost of broadcasting n items of data between p processors is estimated by

Table 1: Experimental investigation of the quality of partitioning algorithms

p	$d_{p,Metis}$	$2w_{p,Metis}$	d_{3D}	w_{3D}
2	1.0	6090	1.05	5874
4	1.0	12164	1.19	16884
8	1.0	24162	1.21	34450
12	1.0	32836	1.22	60270

$$B(n,p) = R(p)(\alpha_b + \beta_b n), \tag{12}$$

where $R(p)$ depends on the algorithm used to implement the *broadcast* operation and the architecture of the computer. For the simplest algorithm, $R(p) = p$. Taking into account the time $\mathcal{O}(n)$ required to read data from the file and assuming that $\alpha_b \ll \beta_b n$, a bound on costs of grid initialization is found

$$W_{p,init} = (c_0 + \beta_b p)n. \tag{13}$$

Note that this part of computations does not depend on the number of nonlinear iterations and the number of time steps. Therefore, the initialization costs can be neglected for problems where a long transition time is simulated.

3.2 Parallel Discretization

The sequential algorithm is decomposed into local computations supplemented with corresponding communication operations. The matrixes and right-hand side vectors are assembled element by element. This can be done locally by each processor, if all ghost values of the vectors belonging to overlapping regions are exchanged among processors. The data communication is implemented by an *odd - even* type algorithm and can be done in parallel between different pairs of processors. Thus, we can estimate the costs of data exchange operation as

$$W_{exch} = \alpha_e + \beta_e m,$$

where m is the number of items sent between two processors, α is the message startup time and β is the time required to send one element of data.

The time required to calculate all coefficients of the discrete problem is given by

$$W_{p,coeff} = c_1 d_p \frac{n}{p},$$

where d_p is a load disbalance parameter.

3.3 Parallel BiCGSTAB algorithm

The sequential BiCGSTAB algorithm is modified in a way such that its convergence properties are not changed during the parallelization process. The only exception is due to implementation of the preconditioner D. If D is a diagonal part of the matrix, i.e. for Jacobi smoothing a parallel realization of the preconditioner is exactly the same as for sequential one. In the case of ILU preconditioner, parallelization can reduce the convergence rate of the parallel BiCGSTAB algorithm. These questions will be addressed in the following section. Here, it is assumed that the number of iterations required to solve the systems of linear equations (6), (7) (see sequential SuFiS algorithm) are the same for the sequential and parallel versions of the BiCGSTAB algorithm.

Four different operations of the BiCGSTAB algorithm require different data communications between processors:

1. Vector *saxpy* operations (see steps (11), (15) and (22)) can be computed in parallel, when parameters α, β, ω are given. No communication between processors is needed, since all required data is locally available on each processor. The complexity of all vector *saxpy* operations calculated during one iteration is

$$W_{p,saxpy} = c_2 \, d_p \frac{n}{p} \, .$$

2. Implementation of the matrix – vector multiplication at steps (4), (13) and (20) requires additional information when boundary nodes of the local part of the vector x are updated (note, that these nodes are inner nodes in the global grid). Such information is obtained by exchanging data with neighbour processors in the specified topology of processors. The amount of data depends on the grid stencil, which is used to discretize the PDE model, i.e. on the overlap of local subgrids. The communication step can be done in parallel. After exchange of the ghost elements the multiplication Ax is performed locally on each processor. Taking into account that matrixes are sparse, the complexity

of two matrix-vector multiplications during one iteration is estimated by

$$W_{p,mv} = c_3\, d_p \frac{n}{p} + 2(\alpha_e + \beta_e m).$$

3. The computation of inner products of two vectors at steps (2), (7), (14), (21) and (24) require a global communication of all processors: first all processors compute inner products of local parts of vectors and then these local products are summed up. Different algorithms can be used to implement the global reduction step. In MPI, there exists a special function `MPI_ALLREDUCE`, which computes a sum and distributes it to all processors. It is assumed that MPI library is optimized for each type of super–computer, taking into account specific details of the computer network. The complexity of computation of all inner products and norms during one iteration is estimated as

$$W_{p,dot} = c_4\, d_p \frac{n}{p} + 5R(p)(\alpha_r + \beta_r).$$

   For a simple implementation of `MPI_ALLREDUCE` function, when all processors send their local values to the master processor, which accumulates results and broadcasts the sum to all processors, $R(p) = cp$.

4. The computation of the preconditioner D is done locally by each processor without any communication operation, the complexity of this step is given by

$$W_{p,D} = c_5\, d_p \frac{n}{p}.$$

   The solution of linear systems $Dx = b$ also requires only local computations. Thus the complexity of steps (12), (19) is given by

$$W_{p,D^{-1}} = c_6\, d_p \frac{n}{p}.$$

After summing up all the estimates, the theoretical model of the complexity of the parallel SUFIS algorithm is achieved

$$\begin{aligned} W_p = (c_0 + \beta_b p)n + K\big((c_1 + c_5)d_p \frac{n}{p} + c_7(\alpha_e + \beta_e m(p))\big) \\ + N\big((c_2 + c_6 + c_{dot})d_p \frac{n}{p} + c_8 R(p)(\alpha_r + \beta_r) + c_9(\alpha_e + \beta_e m(p))\big), \end{aligned} \tag{14}$$

where K is the number of steps in the outer loop of SuFiS algorithm, and N is a total number of BiCGSTAB iterations. Note that the initialization

costs (i.e. the first term of the total costs) do not depend on the number of time steps K and they can be neglected in the case when K is a large number.

4 Results of Computational Experiments

The accuracy of the theoretical complexity model developed above was tested experimentally. Computations were performed on IBM SP5 computer at CINECA, Bologna and on Virgo cluster of computers at ITWM, Kaiserslautern. Results of simulations are presented in Figure 6.

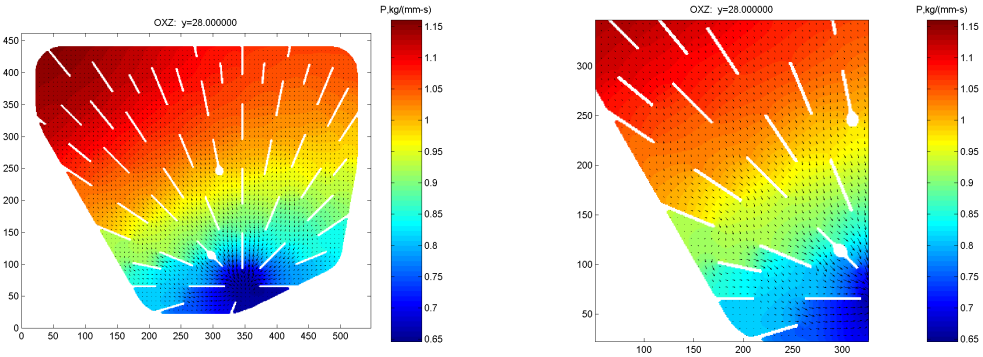

Figure 6: Results of simulations: velocity (arrows) and pressure (colour) in a cross–section of the filter

The same industrial application, as in the example of grid partitioning, is used to test the prediction accuracy of the theoretical model.

In the following tables, experimental times and theoretical predictions of CPU time

$$T_{init} = (c_0 + \beta_b p)n, \quad T_1 = K(c_1 + c_5)d_p\frac{n}{p}, \quad T_2 = N(c_2 + c_6)d_p\frac{n}{p},$$

$$T_{exch} = (Kc_7 + Nc_8)(\alpha_e + \beta_e m), \quad T_{dot} = c_4 d_p\frac{n}{p} + c_8 R(p)(\alpha_r + \beta_r).$$

are presented. For each number of processors, the first line gives experimental values of CPU and the second lines present theoretical predictions. In Table 2, the results of computations are presented for the Virgo cluster of computers.

Table 2: Experimental results and theoretical predictions of CPU times for Virgo cluster

p	T_{init}	T_1	T_2	T_{dot}	T_{exch}	T_{total}	S_p	E_p
1	1.66	106	834	117.3	0.0	1059	1	1
2	3.24	54.5	435	63.0	4.7	560.5	1.89	0.95
	3.24	55.7	438	63.4	4.7	565.0		
4	4.23	31.0	252.5	36.1	6.6	330.7	3.20	0.80
	4.80	31.5	248	37.0	6.1	327.7		
8	5.12	15.6	126.2	20.7	7.3	175.0	6.05	0.76
	6.50	16.0	126.1	21.0	6.4	176.0		
12	7.39	10.7	84.9	15.3	5.1	123.3	8.59	0.71
	9.21	10.8	84.8	15.8	7.3	126.0		

The presented results show that theoretical complexity model gives accurate predictions of different parts of SuFiS algorithm. The efficiency of the parallel algorithm is also good. We note that cluster Virgo uses a Myrinet communication network, therefore communication costs do not reduce seriously the efficiency of the algorithm for a given number of processors.

Results of calculations done on SP5 computer are presented in Table 3.

It can be seen that the theoretical complexity model overestimates the CPU time. The accuracy of the model can be increased if the well-known fact that efficiency of vector operations increases is taken into account. Also, a superlinear speedup of the parallel algorithm is obtained for larger numbers of processors due to the better cache memory utilization in SP5 processors. A simple test was implemented, where matrix operations $A := A + B$, $C := C - D$ were performed many times. The dimension of matrices were taken to be $4 \cdot 10^6$. The following results were obtained:

$$T_1 = 35.3, \;\; T_2 = 15.3, \;\; T_4 = 7.18, \;\; T_8 = 2.83, \;\; T_{16} = 1.29.$$

Data distributon using METIS library

In previous computations, a 3D data decomposition among processors was used. Since the geometry of a computational region is quite complicated, such a decomposition leads to an imbalance of the work-load between processors (up to 1.20 times). Additionally, a general grid distribution algorithm

Table 3: Experimental results and theoretical predictions of CPU times for SP5

p	T_{init}	T_1	T_2	T_{dot}	T_{exch}	T_{total}	S_p	E_p
1	1.83	82.1	735	70.3	0.0	889	1	1
2	3.12	44.1	381	35.7	0.58	464	1.92	0.96
	3.12	43.1	386	37.1	0.58	470		
4	3.58	23.3	203	19.1	3.20	252	3.52	0.88
	5.16	24.4	218.7	21.4	1.2	271		
8	4.54	11.8	93.2	10.2	3.06	122.7	7.25	0.91
	6.66	12.4	111	11.6	1.9	143.6		
12	6.65	8.4	63.0	6.61	3.1	87.8	10.1	0.84
	8.78	8.2	73.8	8.1	2.7	101.5		

was implemented, which was based on graph distribution algorithms implemented in METIS library [23]. In Table 4, the results of computations are presented and the two strategies of data distribution are compared. The computations were performed on Virgo cluster of computers, but in this case a Gigabit Ethernet network is used. The code was compiled with the full optimization option $O3$ in order to make a ration between computation and communication speeds more challenging. In order to get more realistic estimates of the speed-up and the efficiency coefficients, the initialization time T_{init} was excluded from the computation time T_p , since this time could be neglected for real simulations. During the computational experiments, the solution was computed only for six time steps.

Table 4: Experimental results for 3D and METIS data distributions

p	$T_{p,3D}$	$S_{p,3D}$	$E_{p,3D}$	$T_{p,M}$	$S_{p,M}$	$E_{p,M}$
1	885.7	1.00	1.00	893.0	1.00	1.00
2	480.4	1.84	0.92	459.7	1.94	0.97
4	270.6	3.27	0.82	234.3	3.81	0.95
8	150.8	5.87	0.73	127.0	7.03	0.88
12	104.4	8.48	0.70	90.2	9.92	0.83

4.1 Parallel Preconditioners

There have been many studies of the use of various ordering techniques to overcome the trade–off between parallelism and convergence in ILU factorization. Some new multicolor orderings are proposed by D'Azevedo *et al.* [2], Doi and Washio [11], Monga–Made and Van der Vorst [29], Čiegis [9]. The comparison of parallel preconditioners for non-symmetric sparse linear systems is done by Ma [28].

A simple parallel version of ILU preconditioner was implemented, with each processor computing the required factorization by using only a local part of the matrix A. Such a Jacobi type ILU preconditioner is fully parallel, but the convergence rate of the obtained iterations is decreased, see also [3, 9]. It is very difficult to estimate the convergence rate of the BiCGSTAB algorithm with the Jacobi ILU preconditoner even for problems obtained after discretization of the Laplace equation on uniform grids. The efficiency of preconditioners also depends strongly on the given problem coefficients and the properties of the grid. Therefore mainly experimental investigations are used in for the analysis of simplified preconditioners. In Table 5, the performance of BiCGSTAB iterative algorithm with the Jacobi ILU preconditioner is given. Here, N_p is the total number of BiCGSTAB iterations calculated in solving all systems of linear equations by using p processors, and S_p and E_p are the speed-up and efficiency coefficients, respectively, of the parallel SuFiS algorithm. Note that the number of iterations was exactly the same for any number of processors in the previous experiments, i.e. the iterative process was always stopped after computing the maximal number of iterations. The computations were done on the ITWM cluster Virgo.

Table 5: Iteration numbers N_p, CPUtime T_p, speed-up S_P and efficiency E_p coefficients for the Jacobi ILU preconditioner

p	N_p	T_p	S_p	E_p
1	3304	1246	1.00	1.000
2	3741	742	1.68	0.840
4	4070	465.5	2.68	0.670
8	4137	248.6	5.01	0.627
12	4181	175.6	7.10	0.591

It can be seen that the number of iterations for the BiCGSTAB algorithm with the Jacobi ILU preconditioner increase when compared with the global

ILU preconditioner. Therefore, the efficiency of parallel SuFiS is decreased (compare the new values of E_p with the values given in Table 2). However, the quality of the Jacobi ILU preconditioner is still quite satisfactory.

The calculations of the ILU factorization and the solution of problems $Dx = f$ are quite costly. The efficiency of the sequential ILU preconditioner with the Jacobi diagonal preconditioner was compared. The same problem was solved by using $N = 9849$ iterations and $T_1 = 2315$ CPU time. Thus, the number of iterations increased 2.98 times, but the CPU time increased by 1.85 times only.

5 Conclusions

A parallel algorithm for the solution of mathematical models describing filtering of solid particles from liquid oil in complicated 3D geometries is described. Governing Navier-Stokes-Brinkmann equations are discretized by finite volume method, taking special care for accurate discretization of the velocity and the pressure on the interface between the pure fluid regions and the porous media region. The parallelization is based on the data decomposition method. The data is distributed among processors by using two approaches. In the first approach, a structured reference grid is distributed using the optimal decomposition topology. In the second one, the general mesh is decomposed using the Metis library. A theoretical model is proposed for the estimation of complexity of the given parallel algorithm. The theoretical and experimental results obtained are in very good agreement, and thus it can be predicted that the proposed parallel algorithm scales well, and it can be used efficiently for simulation of oil filters with complicated 3D geometries.

Acknowledgements

The work has been partially supported by EC under the project INTAS-30-50-4395, by the Kaiserslautern Excellence Cluster Dependable Adaptive Systems and Mathematical Modelling, DASMOD, and by Eureka Project EUREKA E!3691 OPTCABLES.

References

[1] Angot, Ph.: Analysis of singular perturbations on the Brinkman problem for fictitious domain models of viscous flows. Math. Meth. Appl. Sci., **22** (1999) 1395–1412.

[2] E.F. D'Azevedo, P.A. Forsyth, W.Tang. Ordering methods for preconditioned conjugate gradient methods applied to unstructured grid problems. *SIAM J. Matrix Anal. Appl.*, **10**(4), 567–584, 1992.

[3] Balay, S., K. Buschelman, V. Eijkhout, W.D. Gropp, D. Kaushik, M.G. Knepley, L. Curfman McInnes, B.F. Smith, H. Zhang. PETSc Users Manual. ANL-95/11 - Revision 2.3.0, Argonne National Laboratory, 2005.

[4] R. Barrett, M. Berry, T.F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. SIAM, Philadelphia, PA, 1994.

[5] P. Bastian. *Numerical Computation of Multiphase Flows in Porous Media*. Habilitation Dissertation, Kiel university, 1999.

[6] Bear, J., Bachmat, Y.: Introduction to Modeling of Transport Phenomena in Porous Media. Kluwer Academic Publishers, Dordrecht etc., 1990.

[7] A. Bruaset, A. Tveito. *Numerical Solution of Partial Differential Equations on Parallel Computers*. In: Lecture notes in computer science and engineering. Vol. 51, Springer, 2006.

[8] A. Churbanov, A. Pavlov, P.Vabishchevich, Operator-splitting methods for the incompressibel Navier-Stokes equations on non-staggered grids. I: First-order schemes. *Int. J. Numer. Methods Fluids* **21**, No.8, pp.617-640 (1995).

[9] R. Čiegis. Analysis of Parallel Preconditioned Conjugate Gradient Algorithms. *Informatica*, **16**(3), 317–332 (2005).

[10] R. Čiegis, V.Starikovičius. Realistic performance tool for the parallel block LU factorization algorithm. *Informatica*, **14**(2), 167–180, 2003.

[11] S. Doi, T. Washio. Ordering strategies and related techniques to overcome the trade–off between parallelism and convergence in incomplete factorizations. *Parallel Computing*, **25**, 1995–2014, 1999.

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

[13] Fletcher, C.A.J.: Computational techniques for fluid dynamics. Springer, Berlin etc., 1991.

[14] P.Gresho, R.Sani, *Incompressible flow and the finite element method. Advection-diffusion and isothermal laminar flow.* In collaboration with M. S. Engelman. Chichester: Wiley. 1044 p. (1998).

[15] Fraunhofer Institute for Industrial Mathematics, Annual Report, 2002. http://www.itwm.fhg.de/en/zentral–berichte/annualreport2002/

[16] Fraunhofer Institute for Industrial Mathematics, Department Flows and Complex Structures, http://www.itwm.fhg.de/en/sks/indexsks/

[17] A. Gupta, V. Kumar, A. Sameh. Performance and scalability of preconditioned conjugate gradient methods on parallel computers. *IEEE Transactions on Parallel and Distributed Systems*, **6** (5), 455–469, 1997.

[18] R. Hockney. Performance parameters and benchmarking on supercomputers. *Parallel Computing*, **17**, 1111–1130, 1991.

[19] Iliev, O., Laptev, V.: On numerical simulation of flow through oil filters. Comput. Vis. Sci., **6** (2004) 139–146.

[20] Iliev, O., Laptev, V., Vassileva, D.: Algorithms and software for computer simulation of flow through oil filters. Proc. FILTECH Europa, 2003, Düsseldorf, pp.327–334.

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

[22] O. Iliev, D. Stoyanov: Multigrid adaptive local refinement solver for incompressible flows, Fraunhofer ITWM report No.54, 2003

[23] G. Karypis, V. Kumar. Parallel multilevel k-way partitioning scheme for irregular graphs. *SIAM Review*, **41**(2), 278–300, 1999.

[24] Kaviany, M.: Principles of Heat Transfer in Porous Media. Springer, New York etc., 1991.

[25] V. Kumar, A. Grama, A. Gupta, G. Karypis. *Introduction to parallel computing: design and analysis of algorithms.* Benjamin/Cummings, Redwood City, 1994.

[26] H.P. Langtangen. *Computational Partial Differential Equations. Numerical Methods and Diffpack Programming.* Springer, Berlin, 2002.

[27] V. Laptev, Numerical solution of coupled flow in plain and porous media. PhD thesis, Technical University of Kaiserslautern, 2003.

[28] S. Ma. Comparisons of the parallel preconditioners for large nonsymmetric sparse linear systems on a parallel computer. *International Journal of High Speed Computing*, **12**(1), 55–68, 2004.

[29] M. Monga–Made, H.A. Van der Vorst. A generalized domain decomposition paradigm for parallel incomplete LU factorization preconditionings. *Future Generation Computer Systems*, **17**, 925–932, 2001.

[30] P. Vabishchevich, *The method of fictitious domains in problems of mathematical physics. (Metod fiktivnykh oblastej v zadachakh matematicheskoj fiziki.)*, Moskva: Izdatel'stvo Moskovskogo Univ. 158 p. ,1991.

[31] Samarskii, A.A., Vabischevich, P. N., Computional Heat Transfer, Vol. 1-2, John Wiley & Sons Inc., New York, 1995.

[32] S. Turek, *Efficient solvers for incompressible flow problems. An algorithmic and computational approach.* Lecture Notes in Computational Science and Engineering. 6. Berlin: Springer. 352 p. (1999).

Published reports of the Fraunhofer ITWM

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

1. D. Hietel, K. Steiner, J. Struckmeier
A Finite - Volume Particle Method for Compressible Flows
(19 pages, 1998)

2. M. Feldmann, S. Seibold
Damage Diagnosis of Rotors: Application of Hilbert Transform and Multi-Hypothesis Testing
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
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
(23 pages, 1998)

5. A. Klar, R. Wegener
A hierarchy of models for multilane vehicular traffic
Part I: Modeling
(23 pages, 1998)

Part II: Numerical and stochastic investigations
(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
(24 pages, 1998)

7. I. Choquet
Heterogeneous catalysis modelling and numerical simulation in rarified gas flows Part I: Coverage locally at equilibrium
(24 pages, 1998)

8. J. Ohser, B. Steinbach, C. Lang
Efficient Texture Analysis of Binary Images
(17 pages, 1998)

9. J. Orlik
Homogenization for viscoelasticity of the integral type with aging and shrinkage
(20 pages, 1998)

10. J. Mohring
Helmholtz Resonators with Large Aperture
(21 pages, 1998)

11. H. W. Hamacher, A. Schöbel
On Center Cycles in Grid Graphs
(15 pages, 1998)

12. H. W. Hamacher, K.-H. Küfer
Inverse radiation therapy planning - a multiple objective optimisation approach
(14 pages, 1999)

13. C. Lang, J. Ohser, R. Hilfer
On the Analysis of Spatial Binary Images
(20 pages, 1999)

14. M. Junk
On the Construction of Discrete Equilibrium Distributions for Kinetic Schemes
(24 pages, 1999)

15. M. Junk, S. V. Raghurame Rao
A new discrete velocity method for Navier-Stokes equations
(20 pages, 1999)

16. H. Neunzert
Mathematics as a Key to Key Technologies
(39 pages (4 PDF-Files), 1999)

17. J. Ohser, K. Sandau
Considerations about the Estimation of the Size Distribution in Wicksell's Corpuscle Problem
(18 pages, 1999)

18. E. Carrizosa, H. W. Hamacher, R. Klein, S. Nickel
Solving nonconvex planar location problems by finite dominating sets
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
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
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
(30 pages, 2001)

22. D. Hietel, M. Junk, R. Keck, D. Teleaga
The Finite-Volume-Particle Method for Conservation Laws
(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
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
(44 pages, 2001)

25. J. Kuhnert, S. Tiwari
Grid free method for solving the Poisson equation
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
Keywords: Melt spinning, fiber model, Lattice Boltzmann, CFD
(19 pages, 2001)

27. A. Zemitis
On interaction of a liquid film with an obstacle
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
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
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
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
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
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
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

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

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

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

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

Keywords: lattice Boltzmann equation, boudary condistions, bounce-back rule, Navier-Stokes equation
(72 pages, 2002)

39. R. Korn

Elementare Finanzmathematik

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

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

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

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

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

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

Keywords: multiscale structures, asymptotic homogenization, strength, fatigue, singularity, non-local conditions
(14 pages, 2003)

46. P. Domínguez-Marín, P. Hansen, N. Mladenovi´c , S. Nickel

Heuristic Procedures for Solving the Discrete Ordered Median Problem

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

Keywords: discrete location, Integer programming
(41 pages, 2003)

48. S. Feldmann, P. Lang

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

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

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

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

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

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

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

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

Keywords: Two-phase flow in porous media, various formulations, global pressure, multiphase mixture model, numerical simulation
(30 pages, 2003)

56. P. Lang, A. Sarishvili, A. Wirsen

Blocked neural networks for knowledge extraction in the software development process

Keywords: Blocked Neural Networks, Nonlinear Regression, Knowledge Extraction, Code Inspection
(21 pages, 2003)

57. H. Knaf, P. Lang, S. Zeiser

Diagnosis aiding in Regulation Thermography using Fuzzy Logic

Keywords: fuzzy logic,knowledge representation, expert system
(22 pages, 2003)

58. M. T. Melo, S. Nickel, F. Saldanha da Gama

Largescale models for dynamic multicommodity capacitated facility location

Keywords: supply chain management, strategic planning, dynamic location, modeling
(40 pages, 2003)

59. J. Orlik

Homogenization for contact problems with periodically rough surfaces

Keywords: asymptotic homogenization, contact problems
(28 pages, 2004)

60. A. Scherrer, K.-H. Küfer, M. Monz, F. Alonso, T. Bortfeld

IMRT planning on adaptive volume structures – a significant advance of computational complexity

Keywords: Intensity-modulated radiation therapy (IMRT), inverse treatment planning, adaptive volume structures, hierarchical clustering, local refinement, adaptive clustering, convex programming, mesh generation, multi-grid methods
(24 pages, 2004)

61. D. Kehrwald

Parallel lattice Boltzmann simulation of complex flows

Keywords: Lattice Boltzmann methods, parallel computing, microstructure simulation, virtual material design, pseudo-plastic fluids, liquid composite moulding
(12 pages, 2004)

62. O. Iliev, J. Linn, M. Moog, D. Niedziela, V. Starikovicius

On the Performance of Certain Iterative Solvers for Coupled Systems Arising in Discretization of Non-Newtonian Flow Equations

Keywords: Performance of iterative solvers, Preconditioners, Non-Newtonian flow
(17 pages, 2004)

63. R. Ciegis, O. Iliev, S. Rief, K. Steiner

On Modelling and Simulation of Different Regimes for Liquid Polymer Moulding

Keywords: Liquid Polymer Moulding, Modelling, Simulation, Infiltration, Front Propagation, non-Newtonian flow in porous media
(43 pages, 2004)

64. T. Hanne, H. Neu

Simulating Human Resources in Software Development Processes

Keywords: Human resource modeling, software process, productivity, human factors, learning curve
(14 pages, 2004)

65. O. Iliev, A. Mikelic, P. Popov

Fluid structure interaction problems in deformable porous media: Toward permeability of deformable porous media

Keywords: fluid-structure interaction, deformable porous media, upscaling, linear elasticity, stokes, finite elements
(28 pages, 2004)

66. F. Gaspar, O. Iliev, F. Lisbona, A. Naumovich, P. Vabishchevich

On numerical solution of 1-D poroelasticity equations in a multilayered domain

Keywords: poroelasticity, multilayered material, finite volume discretization, MAC type grid
(41 pages, 2004)

67. J. Ohser, K. Schladitz, K. Koch, M. Nöthe

Diffraction by image processing and its application in materials science

Keywords: porous microstructure, image analysis, random set, fast Fourier transform, power spectrum, Bartlett spectrum
(13 pages, 2004)

68. H. Neunzert

Mathematics as a Technology: Challenges for the next 10 Years

Keywords: applied mathematics, technology, modelling, simulation, visualization, optimization, glass processing, spinning processes, fiber-fluid interaction, trubulence effects, topological optimization, multicriteria optimization, Uncertainty and Risk, financial mathematics, Malliavin calculus, Monte-Carlo methods, virtual material design, filtration, bio-informatics, system biology
(29 pages, 2004)

69. R. Ewing, O. Iliev, R. Lazarov, A. Naumovich

On convergence of certain finite difference discretizations for 1D poroelasticity interface problems

Keywords: poroelasticity, multilayered material, finite volume discretizations, MAC type grid, error estimates
(26 pages,2004)

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

On Efficient Simulation of Non-Newtonian Flow in Saturated Porous Media with a Multigrid Adaptive Refinement Solver

Keywords: Nonlinear multigrid, adaptive renement, non-Newtonian in porous media
(25 pages, 2004)

71. J. Kalcsics, S. Nickel, M. Schröder

Towards a Unified Territory Design Approach – Applications, Algorithms and GIS Integration

Keywords: territory desgin, political districting, sales territory alignment, optimization algorithms, Geographical Information Systems
(40 pages, 2005)

72. K. Schladitz, S. Peters, D. Reinel-Bitzer, A. Wiegmann, J. Ohser

Design of acoustic trim based on geometric modeling and flow simulation for non-woven

Keywords: random system of fibers, Poisson line process, flow resistivity, acoustic absorption, Lattice-Boltzmann method, non-woven
(21 pages, 2005)

73. V. Rutka, A. Wiegmann

Explicit Jump Immersed Interface Method for virtual material design of the effective elastic moduli of composite materials

Keywords: virtual material design, explicit jump immersed interface method, effective elastic moduli, composite materials
(22 pages, 2005)

74. T. Hanne

Eine Übersicht zum Scheduling von Baustellen

Keywords: Projektplanung, Scheduling, Bauplanung, Bauindustrie
(32 pages, 2005)

75. J. Linn

The Folgar-Tucker Model as a Differetial Algebraic System for Fiber Orientation Calculation

Keywords: fiber orientation, Folgar–Tucker model, invariants, algebraic constraints, phase space, trace stability
(15 pages, 2005)

76. M. Speckert, K. Dreßler, H. Mauch, A. Lion, G. J. Wierda

Simulation eines neuartigen Prüfsystems für Achserprobungen durch MKS-Modellierung einschließlich Regelung

Keywords: virtual test rig, suspension testing, multibody simulation, modeling hexapod test rig, optimization of test rig configuration
(20 pages, 2005)

77. K.-H. Küfer, M. Monz, A. Scherrer, P. Süss, F. Alonso, A. S. A. Sultan, Th. Bortfeld, D. Craft, Chr. Thieke

Multicriteria optimization in intensity modulated radiotherapy planning

Keywords: multicriteria optimization, extreme solutions, real-time decision making, adaptive approximation schemes, clustering methods, IMRT planning, reverse engineering
(51 pages, 2005)

78. S. Amstutz, H. Andrä

A new algorithm for topology optimization using a level-set method

Keywords: shape optimization, topology optimization, topological sensitivity, level-set
(22 pages, 2005)

79. N. Ettrich

Generation of surface elevation models for urban drainage simulation

Keywords: Flooding, simulation, urban elevation models, laser scanning
(22 pages, 2005)

80. H. Andrä, J. Linn, I. Matei, I. Shklyar, K. Steiner, E. Teichmann

OPTCAST – Entwicklung adäquater Strukturoptimierungsverfahren für Gießereien Technischer Bericht (KURZFASSUNG)

Keywords: Topologieoptimierung, Level-Set-Methode, Gießprozesssimulation, Gießtechnische Restriktionen, CAE-Kette zur Strukturoptimierung
(77 pages, 2005)

81. N. Marheineke, R. Wegener

Fiber Dynamics in Turbulent Flows Part I: General Modeling Framework

Keywords: fiber-fluid interaction; Cosserat rod; turbulence modeling; Kolmogorov's energy spectrum; double-velocity correlations; differentiable Gaussian fields
(20 pages, 2005)

Part II: Specific Taylor Drag

Keywords: flexible fibers; k-ε turbulence model; fiber-turbulence interaction scales; air drag; random Gaussian aerodynamic force; white noise; stochastic differential equations; ARMA process
(18 pages, 2005)

82. C. H. Lampert, O. Wirjadi

An Optimal Non-Orthogonal Separation of the Anisotropic Gaussian Convolution Filter

Keywords: Anisotropic Gaussian filter, linear filtering, orientation space, nD image processing, separable filters
(25 pages, 2005)

83. H. Andrä, D. Stoyanov

Error indicators in the parallel finite element solver for linear elasticity DDFEM

Keywords: linear elasticity, finite element method, hierarchical shape functions, domain decom-position, parallel implementation, a posteriori error estimates
(21 pages, 2006)

84. M. Schröder, I. Solchenbach

Optimization of Transfer Quality in Regional Public Transit

Keywords: public transit, transfer quality, quadratic assignment problem
(16 pages, 2006)

85. A. Naumovich, F. J. Gaspar

On a multigrid solver for the three-dimensional Biot poroelasticity system in multilayered domains

Keywords: poroelasticity, interface problem, multigrid, operator-dependent prolongation
(11 pages, 2006)

86. S. Panda, R. Wegener, N. Marheineke

Slender Body Theory for the Dynamics of Curved Viscous Fibers

Keywords: curved viscous fibers; fluid dynamics; Navier-Stokes equations; free boundary value problem; asymptotic expansions; slender body theory
(14 pages, 2006)

87. E. Ivanov, H. Andrä, A. Kudryavtsev

Domain Decomposition Approach for Automatic Parallel Generation of Tetrahedral Grids

Key words: Grid Generation, Unstructured Grid, Delaunay Triangulation, Parallel Programming, Domain Decomposition, Load Balancing
(18 pages, 2006)

88. S. Tiwari, S. Antonov, D. Hietel, J. Kuhnert, R. Wegener

A Meshfree Method for Simulations of Interactions between Fluids and Flexible Structures

Key words: Meshfree Method, FPM, Fluid Structure Interaction, Sheet of Paper, Dynamical Coupling
(16 pages, 2006)

89. R. Ciegis , O. Iliev, V. Starikovicius, K. Steiner

Numerical Algorithms for Solving Problems of Multiphase Flows in Porous Media

Keywords: nonlinear algorithms, finite-volume method, software tools, porous media, flows
(16 pages, 2006)

90. D. Niedziela, O. Iliev, A. Latz

On 3D Numerical Simulations of Viscoelastic Fluids

Keywords: non-Newtonian fluids, anisotropic viscosity, integral constitutive equation
(18 pages, 2006)

91. A. Winterfeld

Application of general semi-infinite Programming to Lapidary Cutting Problems

Keywords: large scale optimization, nonlinear programming, general semi-infinite optimization, design centering, clustering
(26 pages, 2006)

92. J. Orlik, A. Ostrovska

Space-Time Finite Element Approximation and Numerical Solution of Hereditary Linear Viscoelasticity Problems

Keywords: hereditary viscoelasticity; kern approximation by interpolation; space-time finite element approximation, stability and a priori estimate
(24 pages, 2006)

93. V. Rutka, A. Wiegmann, H. Andrä

EJIIM for Calculation of effective Elastic Moduli in 3D Linear Elasticity

Keywords: Elliptic PDE, linear elasticity, irregular domain, finite differences, fast solvers, effective elastic moduli
(24 pages, 2006)

94. A. Wiegmann, A. Zemitis

EJ-HEAT: A Fast Explicit Jump Harmonic Averaging Solver for the Effective Heat Conductivity of Composite Materials

Keywords: Stationary heat equation, effective thermal conductivity, explicit jump, discontinuous coefficients, virtual material design, microstructure simulation, EJ-HEAT
(21 pages, 2006)

95. A. Naumovich

On a finite volume discretization of the three-dimensional Biot poroelasticity system in multilayered domains

Keywords: Biot poroelasticity system, interface problems, finite volume discretization, finite difference method.
(21 pages, 2006)

96. M. Krekel, J. Wenzel

A unified approach to Credit Default Swaption and Constant Maturity Credit Default Swap valuation

Keywords: LIBOR market model, credit risk, Credit Default Swaption, Constant Maturity Credit Default Swap-method.
(43 pages, 2006)

97. A. Dreyer

Interval Methods for Analog Circiuts

Keywords: interval arithmetic, analog circuits, tolerance analysis, parametric linear systems, frequency response, symbolic analysis, CAD, computer algebra
(36 pages, 2006)

98. N. Weigel, S. Weihe, G. Bitsch, K. Dreßler

Usage of Simulation for Design and Optimization of Testing

Keywords: Vehicle test rigs, MBS, control, hydraulics, testing philosophy
(14 pages, 2006)

99. H. Lang, G. Bitsch, K. Dreßler, M. Speckert

Comparison of the solutions of the elastic and elastoplastic boundary value problems

Keywords: Elastic BVP, elastoplastic BVP, variational inequalities, rate-independency, hysteresis, linear kinematic hardening, stop- and play-operator
(21 pages, 2006)

100. M. Speckert, K. Dreßler, H. Mauch

MBS Simulation of a hexapod based suspension test rig

Keywords: Test rig, MBS simulation, suspension, hydraulics, controlling, design optimization
(12 pages, 2006)

101. S. Azizi Sultan, K.-H. Küfer

A dynamic algorithm for beam orientations in multicriteria IMRT planning

Keywords: radiotherapy planning, beam orientation optimization, dynamic approach, evolutionary algorithm, global optimization
(14 pages, 2006)

102. T. Götz, A. Klar, N. Marheineke, R. Wegener

A Stochastic Model for the Fiber Lay-down Process in the Nonwoven Production

Keywords: fiber dynamics, stochastic Hamiltonian system, stochastic averaging
(17 pages, 2006)

103. Ph. Süss, K.-H. Küfer

Balancing control and simplicity: a variable aggregation method in intensity modulated radiation therapy planning

Keywords: IMRT planning, variable aggregation, clustering methods
(22 pages, 2006)

104. A. Beaudry, G. Laporte, T. Melo, S. Nickel

Dynamic transportation of patients in hospitals

Keywords: in-house hospital transportation, dial-a-ride, dynamic mode, tabu search
(37 pages, 2006)

105. Th. Hanne

Applying multiobjective evolutionary algorithms in industrial projects

Keywords: multiobjective evolutionary algorithms, discrete optimization, continuous optimization, electronic circuit design, semi-infinite programming, scheduling
(18 pages, 2006)

106. J. Franke, S. Halim

Wild bootstrap tests for comparing signals and images

Keywords: wild bootstrap test, texture classification, textile quality control, defect detection, kernel estimate, nonparametric regression
(13 pages, 2007

107. Z. Drezner, S. Nickel

Solving the ordered one-median problem in the plane

Keywords: planar location, global optimization, ordered median, big triangle small triangle method, bounds, numerical experiments
(21 pages, 2007)

108. Th. Götz, A. Klar, A. Unterreiter, R. Wegener

Numerical evidance for the non-existing of solutions of the equations desribing rotational fiber spinning

Keywords: rotational fiber spinning, viscous fibers, boundary value problem, existence of solutions
(11 pages, 2007)

109. Ph. Süss, K.-H. Küfer

Smooth intensity maps and the Bortfeld-Boyer sequencer

Keywords: probabilistic analysis, intensity modulated radiotherapy treatment (IMRT), IMRT plan application, step-and-shoot sequencing
(8 pages, 2007)

110. E. Ivanov, O. Gluchshenko, H. Andrä, A. Kudryavtsev

Parallel software tool for decomposing and meshing of 3d structures

Keywords: a-priori domain decomposition, unstructured grid, Delaunay mesh generation
(14 pages, 2007)

111. O. Iliev, R. Lazarov, J. Willems

Numerical study of two-grid preconditioners for 1d elliptic problems with highly oscillating discontinuous coefficients

Keywords: two-grid algorithm, oscillating coefficients, preconditioner
(20 pages, 2007)

112. L. Bonilla, T. Götz, A. Klar, N. Marheineke, R. Wegener

Hydrodynamic limit of the Fokker-Planck-equation describing fiber lay-down processes

Keywords: stochastic dierential equations, Fokker-Planck equation, asymptotic expansion, Ornstein-Uhlenbeck process
(17 pages, 2007)

113. S. Rief

Modeling and simulation of the pressing section of a paper machine

Keywords: paper machine, computational fluid dynamics, porous media
(41 pages, 2007)

114. R. Ciegis, O. Iliev, Z. Lakdawala

On parallel numerical algorithms for simulating industrial filtration problems

Keywords: Navier-Stokes-Brinkmann equations, Finite Volume discretization method, SIMPLE, parallel computing, data decomposition method
(24 pages, 2007)

Status quo: June 2007