Implementation and Performance Analyses of a Highly Efficient Algorithm for Pressure-Velocity Coupling

Implementierung und Untersuchung einer hoch effizienten Methode zur Druck-Geschwindigkeits-Kopplung
Master-Thesis von Fabian Gabel
Tag der Einreichung:

1. Gutachten: Prof. Dr. rer. nat. Michael Schäfer
2. Gutachten: Dipl.-Ing Ulrich Falk

TECHNISCHE UNIVERSITÄT DARMSTADT

Studienbereich CE
FNB

Implementation and Performance Analyses of a Highly Efficient Algorithm for Pressure-Velocity Coupling
Implementierung und Untersuchung einer hoch effizienten Methode zur Druck-Geschwindigkeits-Kopplung

Vorgelegte Master-Thesis von Fabian Gabel

1. Gutachten: Prof. Dr. rer. nat. Michael Schäfer
2. Gutachten: Dipl.-Ing Ulrich Falk

Tag der Einreichung:

Erklärung zur Master-Thesis

Hiermit versichere ich, die vorliegende Master-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den January 17, 2015

(F. Gabel)

Contents

1 Introduction

This thesis is about.

2 Finite Volume Method for Incompressible Flows – Segregated Approach

The purpose of this section is to present the discretization applied to the set of equations (**??**). The applied discretization techniques depend on the different terms of each equation, thus at first every equation will be discretized individually. The finite volume method relies on the discretization of integral equations, which will be derived at the beginning of each subsection that relies on them. Since the system of partial differential equations to be solved always exhibits coupling at least between the dependent variables pressure and velocity a first solution algorithm, namely the *SIMPLE* algorithm addressed to resolve the pressure velocity coupling is introduced. The efficient coupling of the Navier-Stokes equations to the temperature equation is one part of the present thesis and will be addressed in a separate subsection. Furthermore every problem modelled by partial differential equations needs to provide valid boundary conditions. The discretization of those boundary conditions, that are relevant for the present thesis will be presented in their own subsection.

2.1 Discretization of the Mass Balance

Integration of equation (**??**) over the integration domain of a single control volume P yields after the application of Gauss' integration theorem and the additivity of the Riemann integral

$$\iint\limits_{S} u_i n_i \mathrm{d}S = \sum_{f\in\{w,s,b,t,n,e\}} \iint\limits_{S_f} u_i n_i \mathrm{d}S = 0$$

In the present work the mass balance is discretized using the midpoint rule for the surface integrals and linear interpolation of the velocity to to center of mass of the surface. This leads to the following form of the mass balance:

$$\begin{aligned}
\sum_{f\in\{w,s,b,t,n,e\}} u_{i_f} n_{f_i} S_f &= u_{i_w} n_{w_i} S_w + u_{i_e} n_{e_i} S_e + u_{i_s} n_{s_i} S_s + u_{i_n} n_{n_i} S_n + u_{i_b} n_{b_i} S_b + u_{i_t} n_{t_i} S_t \\
&= (\gamma_w u_{i_W} + (1-\gamma_w) u_{i_P}) n_{w_i} S_w + (\gamma_s u_{i_S} + (1-\gamma_s) u_{i_P}) n_{s_i} S_s \\
&\quad + (\gamma_b u_{i_B} + (1-\gamma_b) u_{i_P}) n_{b_i} S_b + (\gamma_t u_{i_T} + (1-\gamma_t) u_{i_P}) n_{t_i} S_t \\
&\quad + (\gamma_n u_{i_N} + (1-\gamma_n) u_{i_P}) n_{n_i} S_n + (\gamma_e u_{i_E} + (1-\gamma_e) u_{i_P}) n_{e_i} S_e \\
&= 0,
\end{aligned}$$

where γ_f for $f \in \{w,e,s,n,b,t\}$ is the geometrical interpolation factor.

2.2 Discretization of the Momentum Balance

The stationary momentum balance integrated over a single control volume P reads as

$$\underbrace{\iint\limits_{S} (\rho u_i u_j) n_j \mathrm{d}S}_{\text{convection term}} - \underbrace{\iint\limits_{S} \left(\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right) n_j \mathrm{d}S}_{\text{diffusion term}} = -\overbrace{\iiint_V \frac{\partial p}{\partial x_i} \mathrm{d}V}^{\text{sourceterm pressure}} - \overbrace{\iiint_V \rho\beta(T-T_0)\,\mathrm{d}V}^{\text{sourceterm temperature}}$$

where the different terms to be addressed individually in the following sections are indicated. Note that the form of this equation has been modified by using Gauss' integration theorem The terms residing on the left will be treated in an implicit manner whereas the terms on the right will be treated explicitly.

2.2.1 Calculation of Mass Flux – Rhie-Chow Interpolation

2.2.2 Linearization and Discretization of the Convective Term

The convective term $\rho u_i u_j$ of the Navier-Stokes equations is the reason for the non-linearity of the equations. In order to deduce a set of linear algebraic equations from the Navier-Stokes equations this term has to be linearized. As introduced in section (REFERENCE), the non linearity will be dealt with by means of an iterative process, the Picard iteration. The part dependent of the non dominant dependent variable therefore will be approximated by its value from the previous iteration as $\rho u_i^{(n)} u_j^{(n)} \approx \rho u_i^{(n)} u_j^{(n-1)}$. Using the additivity of the Riemann integral the first step is to decompose the surface integral into individual contributions from each boundary face of the control volume P

$$\iint\limits_{S} (\rho u_i u_j) n_j \mathrm{d}S = \sum_{f\in\{w,s,b,t,n,e\}} \iint\limits_{S_f} \rho u_i u_j n_j \mathrm{d}S = \sum_{f\in\{w,s,b,t,n,e\}} F^c_{i,f}$$

where $F^c_{i,f} := \iint\limits_{S_f} \rho u_i^{(n)} u_j^{(n-1)} n_j \mathrm{d}S$ is the convective flux of the velocity u_i through the face S_f.

To improve diagonal dominance of the resulting linear system while maintaining the smaller discretization error of a higher order discretization, a blended discretization scheme is applied using a deferred correction. Since due to the non-linearity of the equations to be solved an iterative solution process is needed by all means, the overall convergence doesn't degrade noticeably when using a deferred correction. Blending and deferred correction result in a decomposition of the convective flux into a lower order approximation that is treated implicitly and the explicit difference between the higher and lower order approximation for the same convective flux. Since for coarse grid resolutions the use of higher order approximations may lead to oscillations of the solution which may degrade or even impede convergence the schemes can be blended by a control factor $\eta \in [0,1]$. For simplicity all further derivations are presented for the boundary face S_e. This decomposition then leads to

$$F^c_{i,e} \approx \underbrace{F^{c,l}_{i,e}}_{\text{implicit}} + \eta \underbrace{\left[F^{c,h}_{i,e} - F^{c,l}_{i,e}\right]^{(n-1)}}_{\text{explicit}}.$$

Note that the convective fluxes carrying a l or h as exponent, already have been linearized and discretized. The discretization applied to the convective flux in the present work is using the midpoint integration rule and blends the upwind interpolation scheme with the linear interpolation scheme. Applied to above decomposition one can derive the following approximations

$$\begin{aligned} F^{c,l}_{i,e} &= u_{i,E} \min(\dot{m}_e, 0) + u_{i,P} \max(0, \dot{m}_e) \\ F^{c,h}_{i,e} &= u_{i,E} \gamma_e + u_{i,P}(1-\gamma_e), \end{aligned}$$

where the variable values have to be taken from the previous iteration step $(n-1)$ as necessary and the mass flux $\dot{m}_e$ has been used as result of the linearization process. The results can now be summarized by presenting the convective contribution to the matrix coefficients a_{E,u_i} and a_{P,u_i} and the right hand side b_{P,u_i} which are calculated as

$$a^c_{E,u_i} = \min(\dot{m}_e, 0), \quad a^c_{P,u_i} = \max(0, \dot{m}_e) \tag{1}$$

$$b^c_{P,u_i} = \quad \eta \left(u^{(n-1)}_{i,E)} (\min(\dot{m}_e, 0) - \gamma_e) \right) \tag{2}$$

$$+ \eta \left(u^{(n-1)}_{i,P} (\max(0, \dot{m}_e) - (1-\gamma_e)) \right) \tag{3}$$

2.2.3 Discretization of the Diffusive Term

2.2.4 Discretization of the Source Term

2.2.5 Assembly of Linear Systems – Final Form of Equations

Coefficients of matrices for momentum are identical except in case of different factors for under-relaxation (underrelaxation (Andersson))(when does this happen) for the main diagonal coefficient. Small example in code, then show image of assembled system.

2.3 Discretization of the Generic Transport Equation

2.4 The SIMPLE-Algorithm

2.4.1 Pressure Correction Equation

2.4.2 Characteristic Properties of Projection Methods

Under-relaxation, slow convergence, inner iterations outer iterations, relative tolerances, also talk about staggered and collocated variable positioning

2.4.3 Dependence on Under-Relaxation – The Pressure-Weighted Interpolation Method

Present an approach for Under-Relaxation independent converged solution. Conduct the proof to show it really works. Present the results for different under-relaxation factors

2.4.4 Coupling of Temperature Equation

Explicit coupling through source term in momentum balances (Boussinesq-Approximation)

2.5 Boundary Conditions on Domain and Block Boundaries

Introduce chapter by talking about the nature of partial differential equations (Hackbusch). Always start with a simple implementation for the generic transport equation, then specialize to Navier-Stokes equation.

2.5.1 Dirichlet Boundary Condition

Only talk about dirichlet for velocities not for pressure.

Problematics of outlet boundary conditions

2.5.2 Wall Boundary Condition

Note that there are different approaches. Explain which approach is used and why (memory efficiency)

2.5.3 Block Boundary Condition

Relevant for block structured grids as for the validity of the domain composition.

3 CAFFA Framework

3.1 PETSc Framework

Keep in mind not to copy the manual

3.1.1 About PETSc

Bell Prize, MPI Programming

3.1.2 Basic Data Types

Vec,Mat (Different Matrix Types and Their effect on complex methods)

3.1.3 KSP and PC Objects and Their Usage

Singularities

3.1.4 Profiling

PETSc Log

3.1.5 Common Errors

Optimization, Interfaces, (ROWMAJOR,COLUMNMAJOR), Compiler Errors not helpful, Preallocation vs. Mallocs

3.2 Grid Generation and Conversion

Generation of block structured locally refined grids with non-matching block interfaces, neighbouring relations are represented by a special type of boundary conditions; Random number generator to move grid points within a epsilon neighbourhood while maintaining the grid intact. Show in a graph how preallocation impacts on runtime.

3.3 Preprocessing

Matching algorithm – the idea behind clipper and the used projection technique; alt.: Opencascade. Efficient calculation of values for discretization. Important for dynamic mesh refinement, arbitrary polygon matching, parallelizable due to easier interface

3.4 Implementation Details of CAFFA

3.4.1 MPI Programming Model

Basic idea of distributed memory programming model, emphasize the differences to shared memory model. Have a diagram at hand that shows how CAFFA sequentially works (schedule) and point out the locations where and of which type (global reduce, etc.) communication is, or when synchronization is necessary.

- after each solve
- pressure reference
- error calculation
- gradient calculation

Point out that one should try to minimize the number of this points such that parallel performance stays high. Better to calculate Velocity and Pressure Gradients at once not by seperately calling this routine.

3.4.2 Convergence Control

Explain how the criterion for convergence is met

3.4.3 Modi of Calculation

there are different modi of calculation, (NS segregated, then scalar; NS and Scalar Segregated; NS coupled and Scalar segregated; Fully coupled (wath out with fully coupled, this term seems to have already another meaning)). Note that for comparison of solvers it is crucial to develop programs on the same basis. This establishes comparability.

3.4.4 Indexing of Variables and Treatment of Boundary Values

Describe MatZeroValues and how it is used to simplify the code. Also loose a word on PCREDISTRIBUTE its advantages and downsides. Compliance of PETSc zero based indexing and CAFFA indexing which considers boundary values. Problems with boundary entries

3.4.5 Field Interlacing

Realization through special arrangement of variables and the use of index sets (subvector objects) and/or preprocessor directives. Advantages (there was a paper I cited in my thesis). Note that not all variables are interlaced (Velocities are, but their gradients are not). Great impact on Matrix structure.

3.4.6 Domain Decomposition, Exchange of Ghost Values and Parallel Matrix Assembly

- Ghost values are stored in local representations of the global vector (state the mapping for those entries).
- Matrix coefficients are calculated on one processor and sent to the neighbour.
- Preallocation as crucial aspect for program performance. For the coupled system the matrix is assembled in a 2-3 step process to save memory for coefficients.
- Present a simple method for balancing the matrix related load by letting PETSc take care of matrix distribution.
- Use Spy function of Matlab to visualize the sparse matrices. Point out advantages of calculating coefficients for the neighbouring cells locally (no need to update mass fluxes, geometric data doesn't need to be shared, small communication overhead since processors assemble matrix parts that don't belong to them (visualize)).
- Paradigm: Each time new information is available perform global updates. Advantages of using matrices: Show structure of matrix when using arbitrary matching vs. higher memory requirements vs. better convergence

3.5 Postprocessing

Visualization of Results with Paraview and Tecplot Export matrices as binaries and visualize them using matlab scripts.

4 Verification of CAFFA

Different parts, describe incremental approach, only present final results. Refer to next section for Validation of CAFFA

4.1 Theoretical Discretization Error

present the Taylor-Series Expansion

4.2 Method of Manufactured Solutions

basically sum up the important points of salari's technical report, symmetry of solution/domain/grid is bad point out that mms is not able to detect errors in the physical model Also loose a word or two about discontinuous manufactured solutions

4.3 Exact and Manufactured Solutions for the Navier-Stokes Equations and the Energy Equation

Not always there is an exact solution. Divergence free approach. Presentation of the used manufactured solution. What if solution is not divergence free? Derivation of equations and modifications to continuity equation. analyze the problem of too complicated manufactured solutions. also use temperature dependent density function. Explain why global mass conservation in a discrete sense is important and how it can be achived. Special domain, vanishing manufactured solution or symmetric manufactured solutions if a higher approximation of boundary fluxes is not feasible.

- `http://scicomp.stackexchange.com/questions/6943/manufactured-solutions-for-incompressible-navier-stokes`

- http://link.springer.com/article/10.1007/BF00948290
- http://physics.stackexchange.com/questions/60476/exact-solutions-to-the-navier-stokes-equations
- http://www.annualreviews.org/doi/pdf/10.1146/annurev.fl.23.010191.001111

4.4 Measurement of Error and Calculation of Order

Different error measures (L2-Norm,completeness of function space, consistency etc.)

4.4.1 Testcase on Single Processor on Orthogonal Locally Refined Grid

4.4.2 Testcase on Multiple Processors on Non-Orthogonal Locally Refined Grid

Give a measure of the grid quality.

5 Comparison of Solver Concepts

5.1 Convergence Behaviour on Locally Refined Block Structured Grids with Different Degrees of Coupling

Show how the implicit treatment of block boundaries maintains (high) convergence rates. Plot Residual over number of iterations.

5.2 Parallel Performance

5.2.1 Employed Hardware and Software – The Lichtenberg-High Performance Computer

- Networking
- Mem Section and processes in between islands (calculating across islands)
- Versioning information (PETSc,INTEL COMPILERS,CLIPPER,MPI IMPLEMENTATION,BLAS/LAPACK)
- Software not designed to perform well on desktop PCs.

5.2.2 Measures of Performance

- Maße definieren
- Nochmal Hager,Wellein studieren
- Guidelines for measuring performance (bias through system processes or user interaction), only measure calculation time do not consider I/O in the beginning and the end
- Cite Schäfer and Peric with their different indicators for parallel efficiency, load balancing and numerical efficiency

5.2.3 Preliminary Upper Bounds on Performance – The STREAM Benchmark

Pinning of processes (picture), preliminary constraints by hardware and operating systems, identification of bottlenecks and explain possible workarounds, history and results of STREAM. Bandwidth as Bottleneck, how to calculate a Speedup estimate based on the measured bandwidth. PETSc Implementation of STREAM

5.2.4 Discussion of Results for Parallel Efficiency

5.2.5 Speedup Measurement for Analytic Test Cases

5.3 Test Cases with Varying Degree of Non-Linearity

As Peric says I want to prove that the higher the non-linearity of NS, the better relative convergence rates can be achieved with a coupled solver. Fi

5.3.1 Transport of a Passive Scalar – Forced Convection

5.3.2 Buoyancy Driven Flow – Natural Convection

5.3.3 Flow with Temperature Dependent Density – A Highly Non-Linear Test Case

Maybe I could consider two test cases, one with oscillating density and one with a quadratic polynomial. Interesting would be also to consider the dependence of convergence on another scalar transport equation

5.4 Realistic Testing Scenarios – Benchmarking

Also consider simple load balancing by distributing matrix rows equally

5.4.1 Flow Around a Cylinder 3D – Stationary

Describe Testing Setup (Boundary conditions and grid). Present results and compare them with literature.

5.4.2 Flow Around a Cylinder 3D – Instationary

- `http://www.featflow.de/en/benchmarks/cfdbenchmarking/flow/dfg_flow3d/dfg_flow3d_configuration.html`

Describe Testing Setup (Boundary conditions and grid). Present results and compare them with literature.

5.4.3 Heat-Driven Cavity Flow

- `http://www.featflow.de/en/benchmarks/cfdbenchmarking/mit_benchmark.html`

Describe Testing Setup (Boundary conditions and grid). Present results and compare them with literature.

5.5 Realistic Testing Scenario – Complex Geometry

6 Conclusion and Outlook

Turbulence (turbulent viscosity has to be updated in each iteration), Multiphase (what about discontinuities), GPU-Accelerators, Load-Balancing, dynamic mesh refinement, Counjugate Heat Transfer with other requirements for the numerical grid, grid movement, list some papers here) Identify the optimal regimes / conditions for maximizing performance. Each solver concept has its strengths and weaknesses. Try other variants of Projection Methods like SIMPLEC, SIMPLER, PISO or PIMPLE (OpenFOAM)