A Co-Processing Approach for the Visualization of Turbulent Flows

F. Schäfer, M. Breuer, F. Durst

Institute of Fluid Mechanics (LSTM)
University of Erlangen-Nuremberg
Cauerstr. 4
91058 Erlangen
Germany

Email: frank@lstm.uni-erlangen.de

LRZ-Account: `b2101au`

Abstract

Co-visualization offers a possibility for the reduction of increasingly large flow data sets produced in computational fluid dynamics. The present paper describes a particle tracing module which was integrated into a parallel multiblock flow simulation program within the frame of a co-visualization approach. With respect to the application on supercomputers, parallelization and vectorization of the particle tracer were important requirements. The module was applied to three-dimensional, time-dependent simulations of turbulent flow problems.

Keywords: visualization; co-visualization; particle tracing; computational fluid dynamics; turbulence; direct numerical simulation.

1 Introduction

Due to the progress in high-performance computing and the development of efficient numerical algorithms, today simulations of increasingly complex flow problems are state-of-the-art in computational fluid dynamics (CFD). By solving the governing equations of fluid dynamics numerically, CFD applications provide calculated values of the flow field variables such as velocity and pressure at discrete spatial grid points and time steps. Depending on the spatial and temporal resolution, simulations of three-dimensional time-dependent flows can easily produce several gigabytes of solution data. Moreover, in the extreme case of direct numerical simulations (DNS) of turbulent flows very fine computational grids and small time steps have to be used in order to resolve the smallest vortical flow structures. Typical DNS problem sizes are in the range of about 10^7 grid points and 10^4 time steps, so that the resulting flow field data sets require several terabytes of storage.

In order to store the flow information more efficiently, data reduction techniques are of great interest. One approach for data reduction is to co-process the results of the simulation by integrating visualization partly into the flow simulation program. Such a coupling of flow simulation and visualization is refered to as co-visualization. It offers a potential for a strong reduction of the data output by restricting to visualization data such as particle traces or isosurfaces. This approach has its specific advantages and disadvantages [1, 6] compared to the more common post-processing, where visualization is separated from the simulation. Besides the benefit of data reduction, fast hardware components can be exploited in co-processing, since it ideally takes place on a high-performance computer. Furthermore, co-visualization offers the possibility for monitoring the flow simulation online, by sending the visualization data over a network from the high-performance computer used for simulation to a graphics workstation. However, a serious drawback of co-processing is that the flow simulation has to be repeated when visualization parameters are changed.

This paper describes a particle tracing module which has been integrated into a CFD program within the frame of a co-visualization approach. Details about the CFD program and the particle tracing module are given in Sections 2 and 3, respectively. Parallelization and vectorization of the visualization code are addressed in Section 4. Finally, in Section 5 some applications to flow problems of practical interest are presented.

2 Flow Solver

For the present investigation the general-purpose CFD package FASTEST-3D developed by LSTM Erlangen [2, 3] is applied. Laminar as well as turbulent steady and unsteady flows including heat and mass transfer can be simulated numerically. The three-dimensional incompressible Navier-Stokes equations expressing the conservation of mass, momentum and energy are solved based on a finite volume discretization on non-orthogonal curvilinear grids. In order to resolve complex geometries, block-structured grids are used. For parallelization a grid partitioning technique combined with explicit message passing based on MPI is employed. The code is endowed with fast and efficient numerical algorithms such as FAS/FMG multigrid and highly optimized for high-performance supercomputers such as vector computers and parallel-vector systems. FASTEST-3D has been applied successfully to a long list of engineering and scientific flow problems ranging from external flows around high-speed trains to internal flows in stirred vessels and nearly everything in between.

3 Particle Tracing Module

Particle tracing techniques are widely used in flow field visualization. A tutorial for particle tracing in steady and unsteady flows can be found in [4]. A particle trace is given by the solution of the initial value problem

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{v}(\boldsymbol{x}, t),\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \tag{1}$$

where $\boldsymbol{x}(t)$ is the particle position at time t, $\boldsymbol{v}$ denotes the velocity field and $\boldsymbol{x}_0$ is the initial position of the particle at $t = t_0$. Since the velocity field is given as a CFD solution at discrete positions in space and time, an interpolation scheme has to be applied to approximate the velocity at arbitrary positions.

An important topic in particle tracing on curvilinear grids is whether to approximate a solution of Eq. (1Particle Tracing Moduleequation.86) in physical space (p-space) or in computational space (c-space). In p-space the integration of a particle path takes place on the curvilinear grid, making the location of a given point in the grid and thereby the interpolation of the velocity at this point difficult. Generally an iterative search has to be performed to find the grid cell containing a given point. On the other hand, for tracing particles in c-space, the curvilinear grid is transformed to an orthonormal Cartesian grid, in which point location is quite trivial (compare Fig. 1). However, a loss of accuracy may occur due to transformation errors, discontinuities in the transformed velocity field, and additional interpolation errors. A comparison of several p-space and c-space schemes is presented in [5]. Details about error sources and error minimization in c-space schemes can be found in [6].

Within the scope of the present investigation, a parallel particle tracing module has been implemented as part of the CFD package FASTEST-3D. Since FASTEST-3D is optimized for vector computers, vectorization of the particle tracing scheme is an important issue. For easier vectorization a c-space scheme has been chosen, since p-space schemes are more difficult to vectorize due to the iterative point location in these schemes. Three different c-space algorithms have been investigated with respect to their accuracy and performance. For the numerical integration of Eq. (1Particle Tracing Moduleequation.86) several Runge-Kutta schemes have been implemented. A detailed description of the particle tracing module is given in [6].

4 Parallelization and Vectorization

Parallelization of FASTEST-3D is based on a grid partitioning approach, i.e. different blocks of the grid can be assigned to different processors. The parallelization of the particle tracing module is quite straightforward and

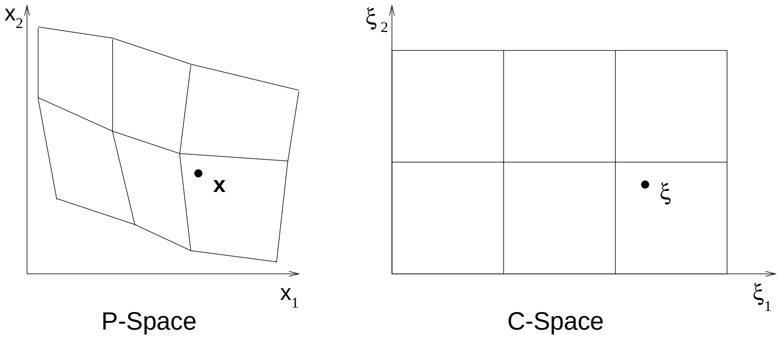

Figure 1: Location of a point within a curvilinear p-space grid and the corresponding orthonormal grid in c-space (2D schematic drawing). The velocities are given at the intersections of the solid grid lines. To interpolate the velocity at some position $\boldsymbol{x}$ within the grid, a search of the grid cell containing $\boldsymbol{x}$ has to be performed. While this has to be done iteratively in p-space (starting from some initial guess), in c-space there is an explicit relation between the grid cell containing the query position and its c-space coordinates $\boldsymbol{\xi}$.

benefits from the parallel structure of the flow solver. Within each block the numerical integration of the particle traces is done by the corresponding processor. When particles are passing the interface between two blocks, they have to be exchanged between the blocks and, if necessary, between the corresponding processors. Here the parallel structure of FASTEST-3D can be used to its full extend. When a particle leaves a block, no search of the destination block has to be performed since it is known a priori due to look-up tables. Furthermore, no search of the particle's new c-space position is necessary since a well defined coordinate transformation can be applied to get the c-space position in the new block from the position in the previous one.

Because no data dependencies or recursions exist within the implemented Runge-Kutta integration schemes, they can be easily vectorized over all particles on one processor. This is mainly due to the explicit point location in c-space.

5 Results and Discussion

The particle tracing module has been applied for the visualization of different flows. Two important applications are presented in this section.

The first application case is a three-dimensional, time-dependent simulation of the turbulent flow in a vessel stirred by a Rushton turbine [7]. A streak line visualization is shown in Fig. 2. Streak lines can be constructed by starting particles at each time step at some fixed seed positions and connecting all particles coming from the same seed position. Since this visualization method is time-dependent, animations are most powerful. However, Fig. 2 shows a snapshot depicting the streak lines at one instant in time. The seed positions are located behind an impeller blade. In order to investigate the vortices being dragged along with the blade, the seed positions are fixed with respect to the blade, that means they are rotating with the same angular velocity. The streak lines reveal two vortices behind the blade, one above and one below the impeller plane. This vortical region is restricted to a small distance behind the blade.

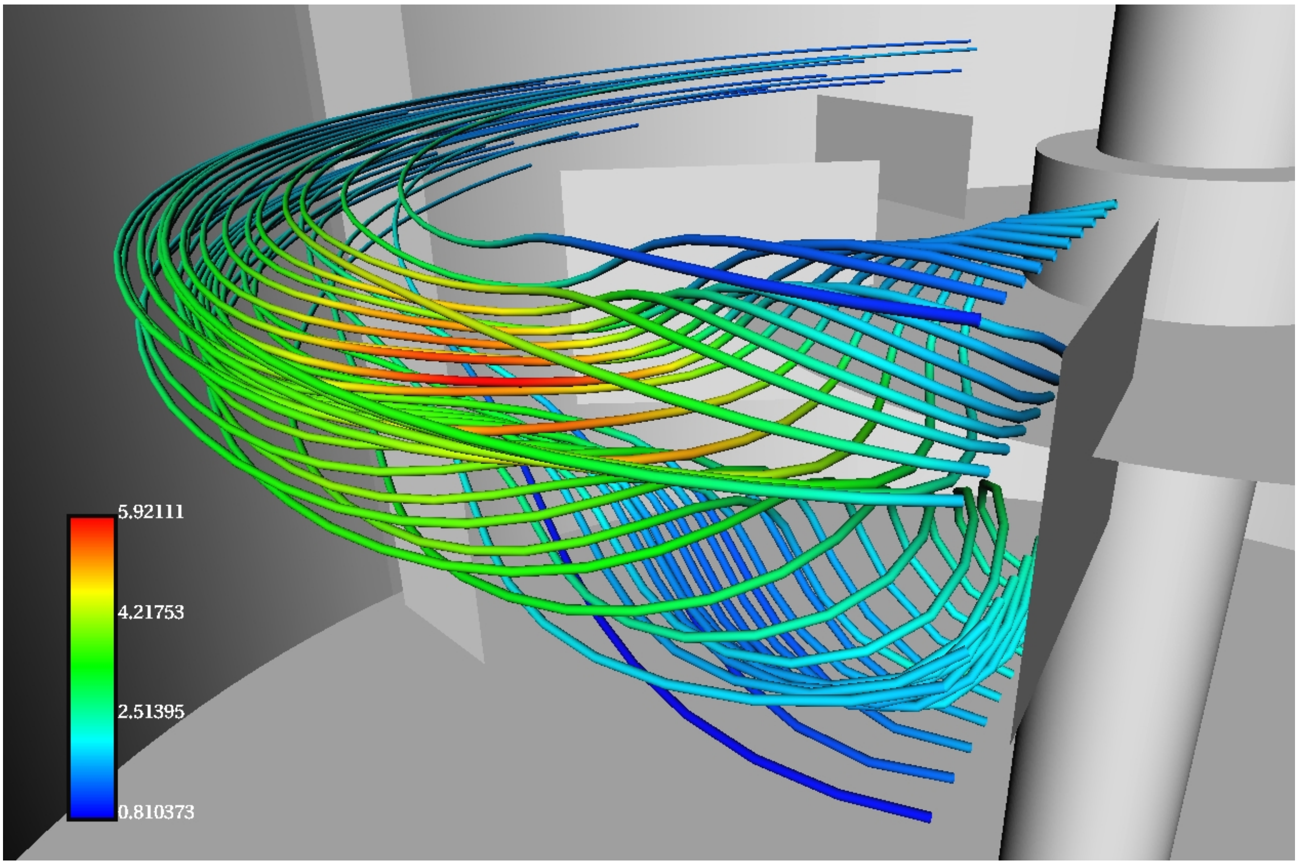

Figure 2: Streak lines started behind a blade of a Rushton turbine (the impeller is rotating counterclockwise). Color coding according to the turbulent kinetic energy has been applied.

The second application case is a direct numerical simulation of the turbulent flow in a channel. The underlying computational grid contains about seven million control volumes. Tracing of particles has been performed over 6350 time steps with 200 particle start positions specified at the inlet of the channel. New particles have been created all 10 time steps at the given start positions. A snapshot of the last time step is depicted in Fig. 3, showing the high degree of turbulence characterizing the flow. The storage requirements for the complete flow solution of this simulation would have been about 1.3 TByte, whereas for the particle tracing data only 5.5 GByte are needed. This leads to a data reduction factor of 236 in this application case.

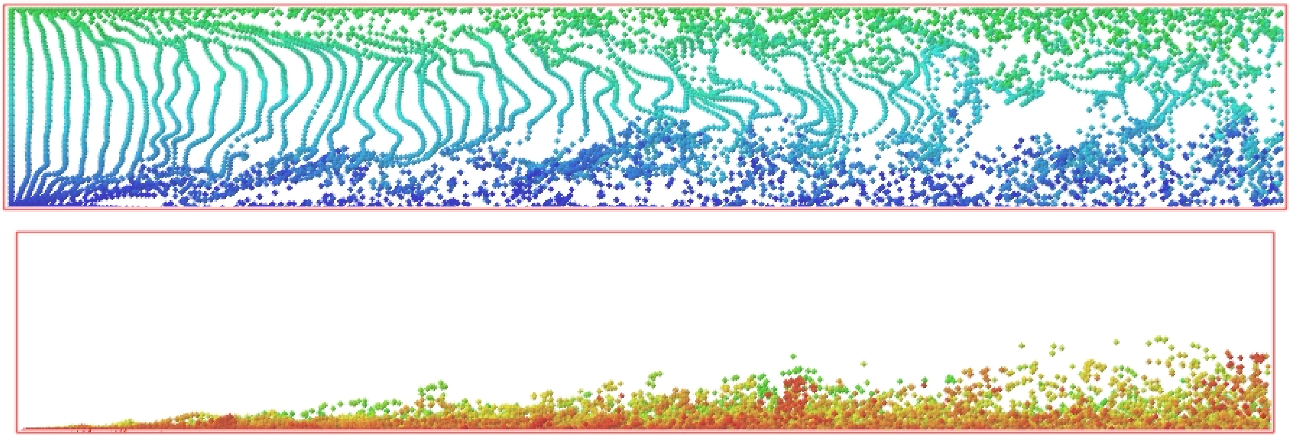

Figure 3: Turbulent flow in a channel, seen from the side. The flow direction is from left to right. Particles are started at the inlet in a vertical rake (top) and in a horizontal rake (bottom) perpendicular to the flow direction. Color coding according to the particle start position has been applied.

6 Publications Arising from this Project

The results of this research project have been presented at the Erlangen Workshop *Vision, Modeling and Visualization '99* [6]. Additional information including video sequences can be found at`http://www.lstm.uni-erlangen.de/SFB`

Bibliography

[1] D. A. Lane, "Visualization of Numerical Unsteady Fluid Flows", *Technical Report NAS-95-017*, NASA Ames Research Center, 1995.

[2] F. Durst, and M. Schäfer, "A Parallel Block-Structured Multigrid Method for the Prediction of Incompressible Flows", *Int. J. Num. Methods Fluids 22*, pp. 549–565, 1996.

[3] F. Durst, M. Schäfer, and K. Wechsler, "Efficient Simulation of Incompressible Viscous Flows on Parallel Computers". In "*Flow Simulation with High-Performance Computers II, Notes on Numerical Fluid Mechanics 52*", pp. 87–101, Vieweg, Braunschweig, 1996.

[4] D. A. Lane, "Scientific Visualization of Large-Scale Unsteady Fluid Flows". In "*Scientific Visualization: Overviews, Methodologies, and Techniques*", pp. 125–145, IEEE Computer Society Press, 1997.

[5] A. Sadarjoen, Th. van Walsum, A. J. S. Hin, and F. H. Post, "Particle Tracing Algorithms for 3D Curvilinear Grids", *Fifth Eurographics Workshop on Visualization in Scientific Computing*, 1994.

[6] F. Schäfer, M. Breuer, "Integrated Particle Tracing within a Parallel Multiblock Flow Simulation Program: Validation and Application", *Vision, Modeling and Visualization '99*, B. Girod, H. Niemann, H.-P. Seidel (eds.), Erlangen, November 17–19, 1999.

[7] F. Durst, M. Schäfer, and K. Wechsler, "*Combined Application of Experimental and Numerical Methods for Stirred Vessel Investigations*", to appear.