10.th Workshop on Two–Phase Flow Predictions, April 2002,
Martin–Luther–Universität Halle–Wittenberg, Halle (Saale), Germany,
Proceedings edited by M. Sommerfeld, pp. 418–423.

Simulation of Unsteady Gas–Particel–Flows including Two– and Four–Way–Coupling on a MIMD Computer Architectur

DI. K. Pachler, Dr.–Ing. Th. Frank, Dr. sc. nat. K. Bernert
Technische Universität Chemnitz, Germany
Fakultät für Maschinenbau und Verfahrenstechnik
Email: klpa@imech.tu–chemnitz.de
http://www.imech.tu–chemnitz.de

1 Introduction

The transport or the separation of particulate matter is a common task in mechanical and process engineering. To improve machinery and physical processes (e.g. for coal combustion, reduction of NO_X and soot) an optimization of complex phenomena by simulation applying the fundamental conservation equations is required. Gas–particle flows are characterized by the ratio of density of the two phases $\gamma = \rho_P/\rho_F$, by the Stokes number $St = \tau_P/\tau_F$ and by the loading in terms of void and mass fraction.

Those numbers (Stokes number, γ) define the flow regime and which relevant forces are acting on the particle. Dependent on the geometrical configuration the particle–wall interaction might have a heavy impact on the mean flow structure. The occurrence of particle–particle collisions is also a question of the local void fraction.

For dilute to moderate dense particle flows the Euler–Lagrange method is capable to resolve the main flow mechanism. An accurate computation needs unfortunately a high number of numerical particles ($1 \cdot 10^5 - 1 \cdot 10^7$) to get the reliable statistics for the underlying modelling correlations. Due to the fact that a Lagrangian algorithm cannot be vectorized for complex meshes the only way to finish those simulations in a reasonable time is the parallization applying the message passing paradigma.

Frank et al. [5],[6],[7] describes the basic ideas for a parallel Euler–Lagrange solver, which uses multigrid for acceleration of the flow equations. The performance figures are quite good, though only steady problems are tackled. For the unsteady case the efficiency is worse, because the $\Delta\tau$ for the time integration along one particle trajectory is very small per one time step of fluid flow integration and so the floating point workload is also very low.
Much time is spent for communication and waiting, because for cold flow particle convection not very extensive calculations are necessary. One remedy might be a highspeed switch like Myrinet or Dolphin PCI/SCI (500 MByte/s), which could balance the relative high floating point performance of INTEL PIII processors and the weak capacity of the Fast–Ethernet communication network(12 MByte/s). Corresponding to the discussed examples calculation times and parallel performance will be presented. Another point is the communication of many small packages, which should be summed up to bigger messages, because each message requires a startup time independently of its size. Summarising the potential of such a parallel algorithm, it will be shown that a Beowulf–cluster is a highly competitve alternative to the classical main frame computer even if heavy number crunching capability is necessary.

2 Mathematical Model

For a turbulent, isotherm, low Mach number gas flow the conservation equations for mass, momentum and scalar species are used. Newton's stress–strain assumption and the standard k-ϵ–model are applied. The PDE can be presented in the following general form.

$$\frac{\partial}{\partial t}\left(\rho_F \Phi\right) + \frac{\partial}{\partial x_j}\left(\rho_F u_j \Phi\right) - \frac{\partial}{\partial x_j}\left(\Gamma \frac{\partial \Phi}{\partial x_j}\right) = S + S^P,$$

while Φ stands for the cartesian velocity components u_i $i = 1 \ldots 3$, for the turbulent kinetic energy k, for the energy dissipation ε and for the scalars c_i.

Equation	Φ	Γ	S	S^P
Mass	1	0	0	$\sum_k \frac{\partial m_k^P}{\partial t}$
Momentum	u_i	μ_{eff}	$\frac{\partial}{\partial x_j}\left(\Gamma \frac{\partial u_j}{\partial x_i}\right) - \frac{\partial p}{\partial x_i} + \rho_F f_i$	$\sum_k \frac{\partial (m_k^P u_{ki}^P)}{\partial t}$
turb. kin. Energy	k	$\frac{\mu_t}{\sigma_k}$	$P_k - \rho_F \varepsilon$	$\overline{S_{u_i}^P \tilde{u}_i} - \overline{S_{u_i}^P} \overline{u_i}$
Dissipation	ε	$\frac{\mu_t}{\sigma_\varepsilon}$	$\frac{\varepsilon}{k}(c_{\varepsilon_1} P_k - c_{\varepsilon_2} \rho_F \varepsilon)$	$c_{\varepsilon_3} \frac{\varepsilon}{k} S_k^P$
Scalar	c_i	$\frac{\mu_t}{\sigma_k}$	0	$\sum_k \frac{\partial m_k^P}{\partial t}$
$P_k = \mu_t \frac{\partial u_i}{\partial x_j}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$; $\mu_{eff} = \mu + \mu_t$, $\mu_t = \rho_F c_\mu \frac{k^2}{\varepsilon}$				
c_μ=0.09, c_{ε_1}=1.44, c_{ε_2}=1.92, σ_k=1.0, σ_ε=1.3, c_{ε_3}=1.1				

Table 1: Diffusion coefficients, production and source terms of the fluid equations

Table 1 shows the variables and source terms for the different equations. Γ is the transport coefficient(Schmidt number), S is the source term of flow variable and S^P is the source term for the interaction with the dispersed phase fulfilling the two–way coupling. μ is the molecular μ_t the turbulent viscosity, ρ_F is the gas density and f_1, f_2 and f_3 are the cartesian components of external forces (e.g. bodyforces) per mass.
m_k^P is the particle mass, whereby the summations index k runs for all particles per cell. u_{ki}^P is the i-th velocity component of the particle k.

The disperse phase is represented by discrete particle trajectories, wheras each particle is a substitute for, depending on the prescribed massflow, a certain number of real particles with the same physical properties. The conservation equations for momentum, heat and mass are discussed in [1]. Those ordinary differential equations are solved by an explicit Runge-Kutta method with fourth order accuracy and automatic time step adjustment.

3 Numerical Method

The conservation equations are discretised within the framework of the finite volume method. All variables are located in the cell-center. The diffusive fluxes are approximated by central difference scheme, the convective terms are blended between central difference and upwind scheme.
The time discretisation, which is based on a parabolian assumption for the derivatives,

$$\left(\frac{d\Phi}{dt}\right)_{n+1} \approx \frac{3\Phi^{n+1} - 4\Phi^n + \Phi^{n-1}}{2\ \Delta t}$$

can be switched between first and second order Euler implicit scheme by a blending factor. n+1 denotes the current time level. Different timestep size is implemented, which can be activated by an adaptive steering control (e.g. particle collision timescales).
The pressure field is deduced from a correction equation following the *SIMPLE* methodology. A geometrical Multigrid for the pressure equation strongly accelerates the solution process.

The particle–particle collision model due to Sommerfeld [8] uses an stochastic impact model generating a virtual impact partner and an impact probability assuming a poisson distribution. The calculation of the collision frequency is done in analogy to the kinetic gas theory.
The wall impact model considers static and dynamic friction and requires the solution of a conservation equation for angular momentum.

The parallelisation of the gas flow is based on a domain decomposition approach using the from the grid generation process still existing ijk–blocks, which are dedicated to certain processors. The block region is the smallest volume region, which can be allocated by the nodes in a arbitrary manner. The grid–block–to–processor arrangement is static.
The Lagrangian solver uses a master–client system, where the master node sends the start data of each particle trajectory to clients on rest for calculation purposes. Furthermore the master node collects the final particle data in the updated start list. So the particles are spread on all client nodes in a full dynamic arrangement, which leads to a more or less uniform work. Also the fluid data must be distributed to the nodes according due to their currently computed particles.

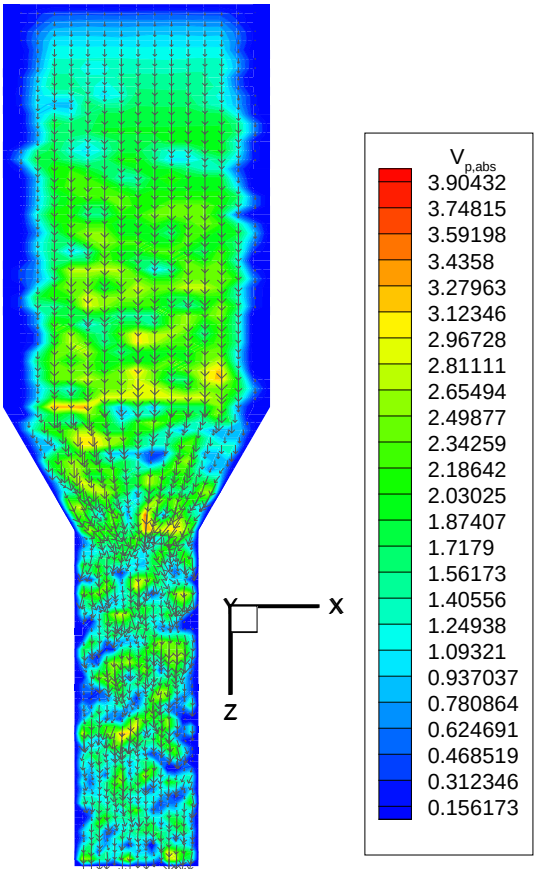

Figure 1: Cell Averaged Particle Velocities in a Convergent Channel

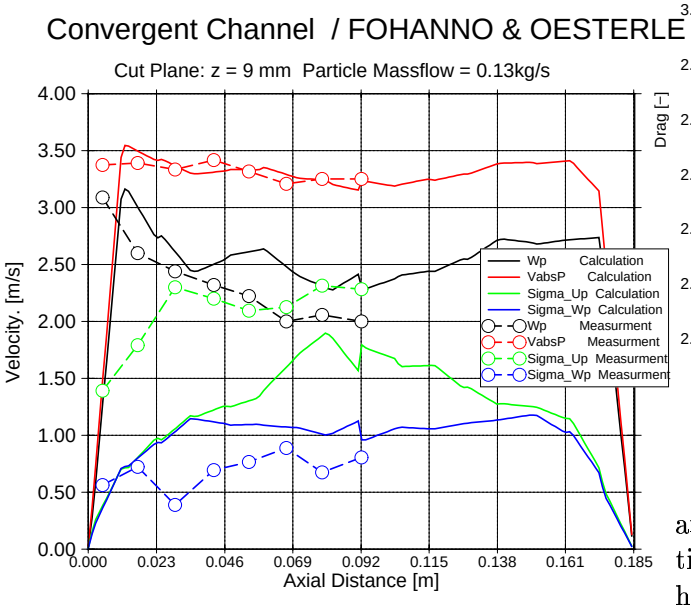

Figure 2: Cell Averaged Mean Particle Velocity/Variance Profiles

4 Results and Validation

To validate the accuracy of the unsteady procedure within *MISTRAL/PartFlow–3D* a DFG-Benchmark [9] was used as a base for comparisons. The task is to judge the quality of the numerics and for this a 2–D and 3–D laminar flow around a cylinder is sufficient. Transient lift and drag coefficients were calculated and fit well with the data of the benchmark as in Table 2 depicted. Measurments [3] were done with LDA to get velocity profiles and out of this the Strouhal number.

Case 2D–2	Dfg–Benchmark	MISTRAL	Measur.
$cd_{Drag_{Max}}$	3.22–3.24	3.22	–
$cd_{Lift_{Max}}$	0.99–1.01	0.986	–
Strouhal No.	0.295–0.305	0.298	0.287±0.003

Table 2: Comparison of Integral Flow Coefficients from DFG-Benchmark

It was pointed out that the second order time discretization is clearly superior to the first order scheme as in figure 3 depicted.

The experiment of Oesterle [4] is used to show

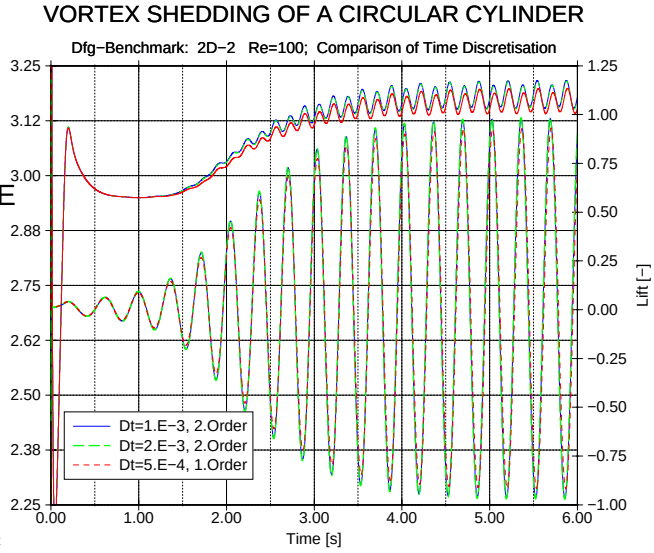

Figure 3: Drag and Lift Coefficient

an example related to high Stokes number particle flow. Two different particle mass flow rates have been measured and calculated. Figure 1 shows the average particle velocity, which reaches at its maximum the particle terminal velocity. Wall interaction and particle–particle collisions lead to a kind of choking in the region of the convergent passage, which also reduces the particle velocity.

Figure 2 shows a comparison of measurment and calculation for mean and fluctuating values of the particle velocity. The measurment plane is located very near the low end of the convergent part of the channel (figure 1).

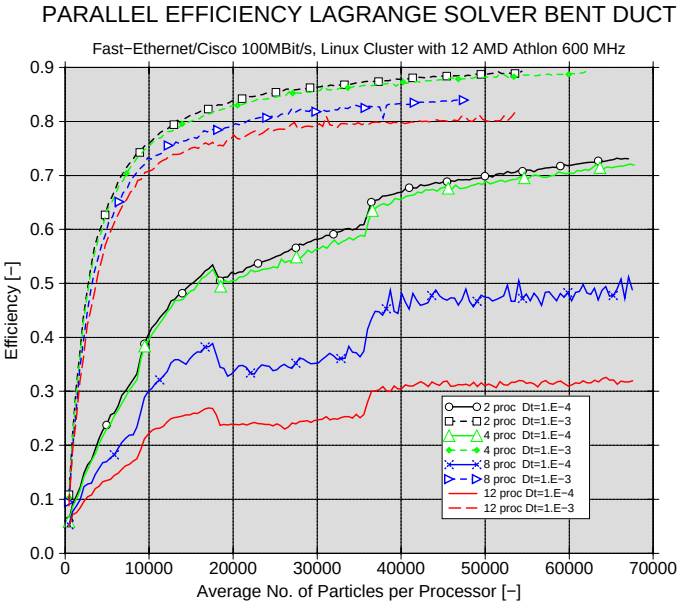

Figure 4: Parallel Efficiency AMD/Cisco Cluster

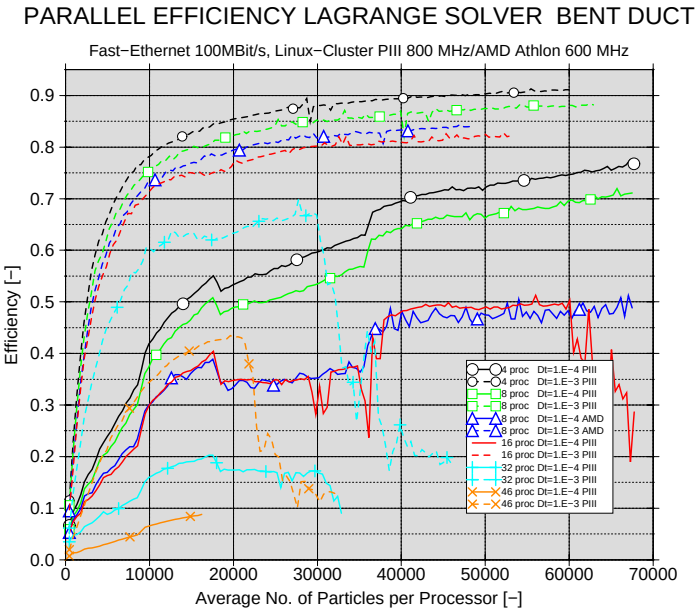

Figure 5: Parallel Efficiency
INTEL/Black Diamond Cluster

Relying on a current ECSC–Project [2] a gaseous pipeflow transporting ceramic particles is displayed in Figure 8. There have been performed experimental investigations measuring velocity profiles with PDA for wide variety of massloadings (m=0.05–2.0). A CFD simulation [10] was done to compare the dilute case. The moder-

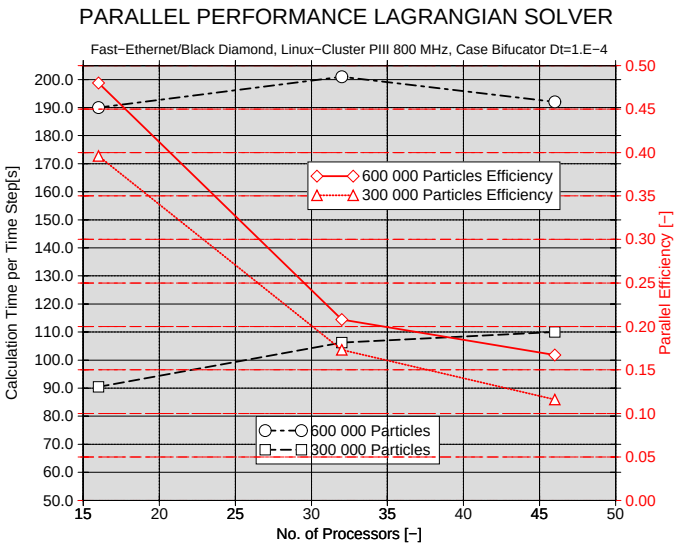

Figure 6: Case Bifucator on
INTEL/Black Diamond Cluster

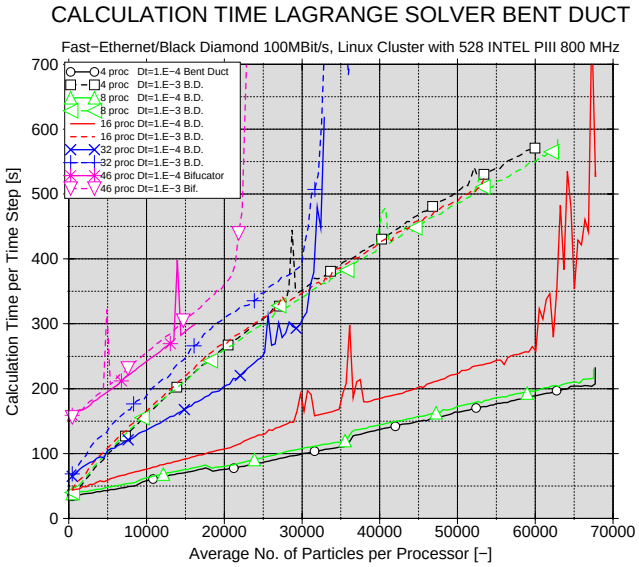

Figure 7: Calculation Time
INTEL/Black Diamond Cluster

ate dense case with massloading m=2.0 shows a very pronounced phase coupling, due to the small Stokes number and the high mass loading. Figure 7 displays the local void fraction and the particle mean velocity. Both quantities are quite different in comparison with the dilute case m=0.05. The particulate phase concentrates behind the first bend at the outside wall of the bent pipe. Within the subsequent straight section a chocking region is built up with locally dense and

dilute regions. After the second curvature the particle phase seems to be spread in a more homogeneous manner, though a dynamic equilibrium is not yet achieved after 1.1 seconds.

The bifucator case and the first curvature of the bifucator named "Bent Duct" are used as test cases for the parallel performance measurments. Two different Beowulf clusters were available for the comparisons (INTEL PIII with Black Diamond switch and an AMD Athlon with Cisco switch based on Fast–Ethernet). Two criteria for parallel performance were checked:
Figure 4 and figure 5 display the efficiency of different cluster sizes for increasing particle load. Figure 5 includes a direct comparison between the INTEL and the AMD cluster. The better performance of the INTEL cluster results mainly from the faster communication network. Figure 7 shows the same cases as Figure 5, but in terms of calculation time. The steep increase of calculation time for high particle numbers (e.g. 46, 32, 16 processors) means, that the master node is in a swapping mode (exceeds the physical memory), because of the very big particle start list. Figure 6 shows efficiency and calculation time for a fixed particle load. Besides two timestep sizes were used (1.E-3 and 1.e-4) to demonstrate their influences on the efficiency of the particle start data exchange.

5 Conclusions

Particulate isothermal flows within the moderate dense flow regime require the consideration of the following phenomena:
Particle–particle collisons, particle wall interactions, particle rotation, two–way coupling, unsteadiness.
A flow simulation in a complex technical geometry asks for a high number of computational particles to get a reasonable accuracy of the computational results. The resolution error scales with $1/\sqrt{n}$ with n = number of particles per control volume. To performe such expensive simulations efficiently the parallelisation of the Lagrangian solver is one solution. The results of a bifucator flow show the dependance of the parallel efficiency on the local flow regime, which determines the gas flow time step size and the frequency of data communication. Generally spoken the steady solver approach is the upper limit of the parallel efficiency. The lower limit depends on the performance of the communication network and the processor speed. There is still some improvement left for optimization of the data exchange procedure within the computational algorithm, though this might require a redesign of the present software.

References

[1] **Crowe C.T., Sommerfeld M., Tsuji Y.:** Multiphase Flows with Droplets and Particles, CRC Press, 1998.

[2] **ECSC Report prepared by Powergen UK plc:** ECSC Coal RTD Programme, Project 7220–PR–050, Techn. Report No. PT/01/BE1001/R, Nottingham Nov. 2001

[3] **Durst, F., Fischer, F., Jovanovic, J., Kikura, H., Lange, C.:** LDA measurments in the wake of a circular cylinder, Flow Simulation with Highperformance Computers II, Notes in Numerical Fluid Dynamics Vieweg Verlag Vol.52, pp. 566–576, Ed. Hirschel, 1996

[4] **Fohanno, S., Oesterle, B.:** Analysis of the Effect of Collisions on the Gravitational Motion of Large Particles in a Vertical Duct, Int.J. of Multi Phase Flows, pp. 267-292, Vol26(2) 2000

[5] **Frank, Th., Bernert, K., Schneider, H., Pachler, K.:** Efficient Parallelization of Eulerian–Lagrangian Approach for Disperse Multiphase Flow Calculations on MIMD Computer Architectures , IEEE International Conference on Cluster Computing, CLUSTER 2000, Nov 28.–Dec 2., 2000, Chemnitz University of Technology, Saxony, Germany

[6] **Frank Th., Bernert K., Pachler K., Schneider H.:** Aspects of Efficient Parallelization of Disperse Gas-Particle Flow Predictions Using Eulerian-Lagrangian Approach, ICMF 2001 - 4th International Conference on Multiphase Flow, Paper No. 311, pp. 1-13, New Orleans,U.S.A., May 27 - June 1, 2001

[7] **Frank Th.:** Parallele Algorithmen für die numerische Simulation dreidimensionaler, disperser Mehrphasenströmungen und deren Anwendung in der Verfahrenstechnik, Habilitation an der Fakultät für Maschinenbau und Verfahrenstechnik der TU Chemnitz, Oktober 2001

[8] **Sommerfeld, M.:** Validation of a stochastic Lagrangian modelling approach for inter–particle collisions in homogeneous isotropic turbulence, J. of Multiphase Flows, 27, pp. 1829–1858, 2001

[9] **Schäfer, M., Turek, S.:** Benchmark Computations of Laminar Flow Around a Cylinder, Preprint 96–03(SFB 359), IWR Heidelberg, 1996

[10] **Schneider, H., Frank, Th., Pachler, K., Bernert, K.:** A numerical study of the gas-particle flow in pipework and flow splitting devices of coal-fired power plant, 10th Workshop on Two–Phase–Flow Predictions,
April 9.–April 12., 2002, Merseburg, Martin–Luther University of Technology, Germany

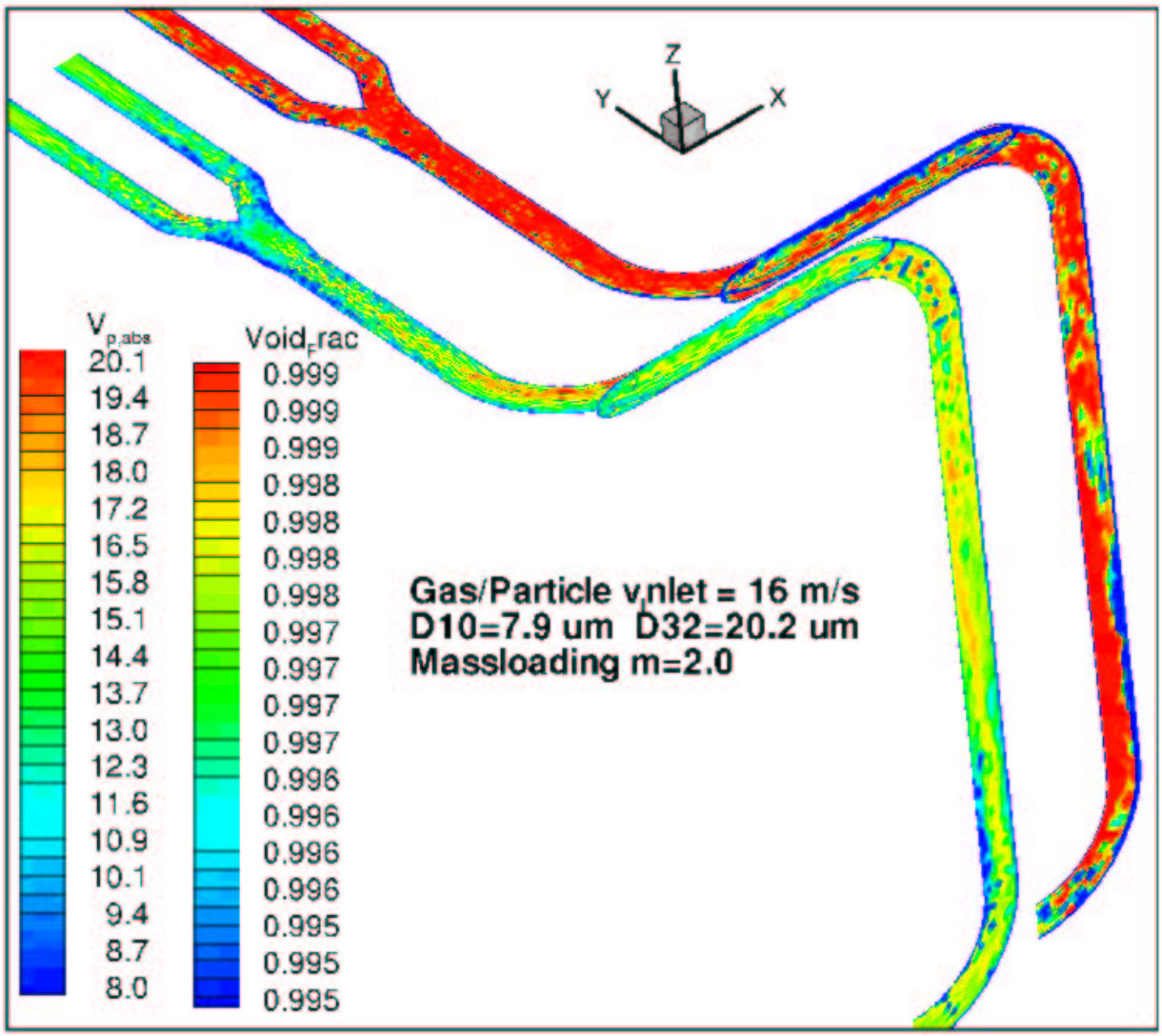

Figure 8: Case Bifucator VoidFraction / Particle Mean Velocity