FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG
INSTITUT FÜR INFORMATIK (MATHEMATISCHE MASCHINEN UND DATENVERARBEITUNG)

Lehrstuhl für Informatik 10 (Systemsimulation)

waLBerla: Exploiting Massively Parallel Systems for CFD

C. Feichtinger, J. Götz, S. Donath, K. Iglberger, U. Rüde

Technical Report 08-2

waLBerla: Exploiting Massively Parallel Systems for CFD

C. Feichtinger, J. Götz, S. Donath, K. Iglberger, U. Rüde

Lehrstuhl für Systemsimulation
Friedrich-Alexander University of Erlangen-Nuremberg
91058 Erlangen, Germany

{christian.feichtinger, jan.goetz, stefan.donath,
klaus.iglberger, ruede}@informatik.uni-erlangen.de

February 14, 2008

The waLBerla project aims at the development of an efficient massively parallel lattice Boltzmann software library providing the necessary features for several computational fluid dynamics applications. In this article we focus on our parallelization of the framework, which is based on a domain partitioning scheme named patch concept. This concept also enables a seamless specialization of the partitions to the different application features as well as the possibility for further optimization such as memory reduction. It is discussed in detail how our design of the patches ensures an efficient and flexible implementation. Furthermore, the communication of the framework is introduced, which includes process-local and global communication. In order to discuss the suitability of the parallelization for massively parallel usage, various test scenarios have been investigated on different architectures. These tests include serial, weak and strong scaling experiments up to 810 cores and up to a domain size of 1530^3 lattice cells.

1 Motivation

In computational fluid dynamics (CFD) many applications of scientific interest share physical and computational aspects. In research environments the usual practice is one program for each application, leading to a reimplementation of the shared physics, the common data structures and also the parallelization, which often requires a considerable effort. Additionally, this replicated functionality has to be validated for each application, again leading to unnecessary work. Hence, there is a need for a fluid simulation library flexible enough to support research for several physical applications that cannot be simulated by existing software packages. The waLBerla software library has been designed to provide such a framework. It further aims at an efficient parallel implementation together with a good usability. For a detailed description of the features of waLBerla e.g. parallel simulation output or input descriptions see [FGD$^+$07]. Most of today's flow simulations are based on numerical schemes that solve the Navier-Stokes (NS) equations directly. However, there exists an alternative approach named lattice Boltzmann method (LBM). This method is based on solving an approximation of the Boltzmann equation and thus is a kinetic-based approach. For the waLBerla software library the LBM has been chosen due to its advantages for the parallelization as well as its suitability for the scheduled applications. These applications cover moving charged colloids [HF01], fluid flow in blood vessels [SGR$^+$07] and multiphase flows through micro porous media in fuel cells. In addition to a design flexible enough to include further applications, the framework has to be suitable for the simulation of large domains, e.g. a representative volume (REV) of the gas diffusion layer (GDL) in a polymer electrolyte fuel cell [Fue07]. The reason for the need of a large domain in the example above is, that the size of the REV[1] is about $0.45 \times 0.45 \times 0.1$ mm^3 and that the volume of a lattice cell has to be $\delta x^3 = 0.1^3 \mu\text{m}^3$, due to accuracy reasons and the limitation of the LBM to small Knudsen Numbers. With a 10% porosity this leads to $1.8 \cdot 10^{10}$ fluid cells, which results in a memory requirement of about 6.5 TB for the LBM (see following paragraph on LBM and [DGF$^+$07] on resource requirements of waLBerla). Such a simulation is not feasible on a single CPU. Hence, the framework has to be adapted for massively parallel architectures.

Therefore one major target is the utilization of the HLRB II which is an SGI Altix 4700 [HLR07] featuring 39 TB memory. A rough estimation shows the power of this machine: With the performance of the current implementation of waLBerla (see Sec. 4.1) the above example results in a theoretical computation time of about 3 hours per time step, given a single core CPU with enough memory. Assuming a parallel efficiency of 70%, a time step would take about 1.5 seconds on the 4864 dual-core CPUs of the HLRB II. Thus

[1] Minimum size of REV based on findings from internal projects. No publications yet.

running fifty thousand time steps would require about 20 hours, instead of 17 years. Hence, only with an efficient parallelization it is possible to simulate the fluid flow in a GDL.

The remainder of this paper is organized as follows: In the subsequent paragraph a brief overview of the LBM is given, followed by the introduction of the waLBerla patch concept in Sec. 2. These patches are subdivisions of the fluid domain, which are the basic components for the parallelization, the optimization strategies, and the flexibility which is needed for the integration of further applications. In Sec. 3 the implementation of the process-local and global communication is explained in detail. Performance results are given in Sec. 4, where the serial performance as well as the parallel performance for various architectures has been evaluated. This first investigation discusses the suitability of the parallel concept for massively parallel usage in basic geometries. The article is concluded in Sec. 5 with a summary and outlook.

Brief Introduction to the Lattice Boltzmann Method

The lattice Boltzmann method is one possible approach to solve CFD problems numerically. It originates from the lattice gas cellular automata (LGCA), whereas McNamara and Zanetti were the first to introduce the Boltzmann collision operator to LGCA in 1988 [MZ88]. Further work [HL97] has shown that the LBM can be directly derived from the continuous Boltzmann equation. Hence, it is independent of the LGCA and based on kinetic theory. It can also be shown that the LBM is equivalent to an explicit finite difference scheme of the Navier-Stokes equations with second order spatial accuracy and first order temporal accuracy [JKL05]. Amongst others the LBM has been successfully applied to free surface flows [KPR$^+$05], multiphase flows [SC93], flows through porous media [ZFB$^+$02], fluid mixtures [LG03, Asi06], blood flows [AHS06] and metal foams [KPR$^+$05].

The advantages of the LBM are the explicit update rule, the fast mesh generation due to the Cartesian grids, and that many macroscopic and hydrodynamic effects result from mesoscopic quantities. A detailed description of the LBM can be found in [Hän04][WG00][YMLS03]. In the remainder of this Section an overview of the governing equations of the LBM is provided.

For the waLBerla software library the D3Q19 stencil [QDL92] and the LBGK [WG00] model are used. With the D3Q19 stencil the LBM is based on cubic cells with 19 unknowns, the particle distribution functions (PDF) $f_\alpha(x_i, t)$, which are defined as the expected amount of particles in the volume δx^3 located at the lattice position x_i with the lattice velocity $e_{\alpha,i}$. The lattice direction α points towards the neighboring cells (see Fig. 1 for an illustration). Discretized in time and space the LBGK model is given in tensor

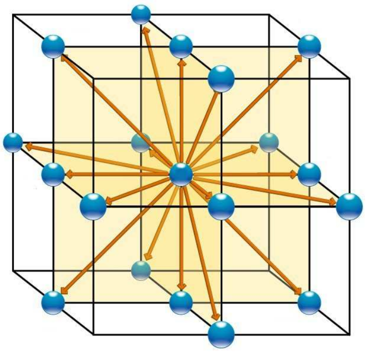

Figure 1: The D3Q19 stencil

notation by:

$$f_\alpha(x_i + e_{\alpha,i}\delta t, t + \delta t) - f_\alpha(x_i, t) = -\frac{\delta t}{\tau}\left[f_\alpha(x_i, t) - f_\alpha^{(eq)}(\rho(x_i, t), u_i(x_i, t))\right]. \tag{1}$$

Due to simplicity quantities depending on x_i and t will be written without their dependencies, e.g. $f_\alpha = f_\alpha(x_i, t)$. Further the relaxation time τ can be determined from the lattice viscosity (Eq. 7). The equilibrium distribution $f_\alpha^{(eq)}(\rho, u_i)$ (Eq. 2) depending on the macroscopic velocity u_i (Eq. 4) and the macroscopic density ρ (Eq. 5) for the isothermal case is given by the Maxwell-Boltzmann distribution function discretized for low mach numbers:

$$f_\alpha^{(eq)}(\rho, u_i) = \rho \cdot w_\alpha \cdot \left[1 + \frac{1}{c_s^2}(e_{\alpha,i} \cdot u_i) + \frac{1}{2c_s^4}(e_{\alpha,i} \cdot u_i)^2 - \frac{1}{2c_s^2}u_i^2\right]. \tag{2}$$

In the D3Q19 model the thermodynamic speed of sound is given by $c_s = \frac{1}{\sqrt{3}}$ and the lattice velocities $e_{\alpha,i}$ and lattice weights w_α are:

$$e_{\alpha,i} = \begin{cases} (0,0,0), \\ (\pm 1,0,0),(0,\pm 1,0),(0,0,\pm 1), \\ (\pm 1,\pm 1,0),(0,\pm 1,\pm 1),(\pm 1,0,\pm 1) \\ \end{cases} \quad w_\alpha = \begin{cases} 1/3, & \alpha = 0 \\ 1/18, & \alpha = 1, 2, 3, 4, 5, 6 \\ 1/36, & \alpha = 7, 8, 9, 10, 11, 12, \\ & \quad 13, 14, 15, 16, 17, 18 \end{cases} . \tag{3}$$