The Lattice-Boltzmann Method for Multiphase Fluid Flow Simulations and Euler-Lagrange Large-Eddy Simulations

J.J. Derksen

Multi-Scale Physics Department, Delft University of Technology, Netherlands

Abstract. Cellular automata for mimicking physical systems, the lattice gas and lattice Boltzmann automata, fluid dynamics with the lattice-Boltzmann method, practicalities of the lattice-Boltzmann method, DNS of solid-liquid suspensions, scaling in single-phase turbulence, direct numerical and large-eddy simulations, subgrid-scale modeling in LES, solid-liquid flow: point particles in LES, passive and reactive scalar transport in turbulent flow, filtered density function approach to turbulent reactive flow.

1 Cellular Automata

In computational science, the challenge is to describe and understand complex systems (that can be loosely defined as systems with many interacting components) by means of efficient numerical tools. Cellular automata turn out to be very fruitful in this respect. Von Neumann introduced the concept of cellular automata (CA) in the 1940’s. The idea is to represent a physical system in terms of discrete space and time. The physical quantities (the state of the automaton) take only a finite set of values.

Von Neumann was intrigued by the mechanisms that lead to self-reproduction in biology. He wanted to devise a system that has the capability to reproduce another system of similar complexity. For this he devised a fully discrete “universe” made of cells. Each cell is characterized by an internal state, typically consisting of a finite number of information bits. The cells evolve in discrete time like simple automata that only know a simple rule to compute their new internal state. The rule determines the evolution of the system. It is the same for all cells and relates the state of a cell with that of its neighbors. Similarly to what happens in biological systems, the activity of cells takes place simultaneously and synchronously.

The *game of life* (proposed by John Conway in 1970) is an example of a simple rule leading to complex behavior. Conway imagined a two-dimensional, square lattice (like a checkerboard). Each cell can be alive (state 1) or dead (state 0). The updating rule is: a dead cell surrounded by exactly three living cells gets back to life; a living cell surrounded by less than two or more than three neighbors dies (of isolation or over-crowdedness). Here sur-

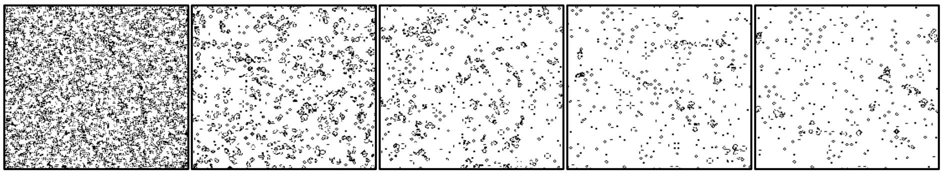

Figure 1. Snapshots of the game of life on a 200x200 grid. Time is running from left to right

rounding means the four nearest neighbors (north, south, east, west), and the four next nearest neighbors (along the diagonal). This very simple automaton has rich behavior. Complex structures appear and show interesting dynamics. In Figure 1 some single realizations of the lattice are given. More interesting, however, are the dynamics of the system that can be viewed in animations.

Another example along the same lines: We define a two-dimensional square lattice with values 0 or 1 on each node. Starting from some initial condition, we evolve the system as follows: if the sum of the four nearest neighbor values is even, the new state is set to zero, if it is odd it is set to 1. When this rule is iterated, nice geometrical patterns are observed. Figure 2 shows results if we start from the situation given in the left panel. The complexity evolving in time results from spatial organization.

Above were artificial (and in a way esoteric) examples. The route towards describing physical systems in terms of a CA is not straightforward. A natural way is to propose a model of what we think is going on. The "art of modeling" is to retain only those ingredients that are essential; the degree of reality of the model depends on our level of description. If we are interested in global, macroscopic properties (and that is mostly the case in fluid dynamics with variables like fluid velocity and pressure), the microscopic details often are not relevant (as long as we obey symmetries and basic conservation laws).

In 1986, Frisch, Hasslacher and Pomeau (Frisch et al., 1986) announced a striking discovery. They showed that the molecular motion need not to be nearly as detailed as real molecular dynamics (with of the order of 10^{23} molecules, and even more degrees of freedom) to give rise to realistic fluid dynamics. Their fluid was constructed of fictitious particles, each with the same mass and moving with the same speed, and differing only in their velocity directions. Moreover, these directions were constrained to a finite set (in two dimensions only six). This was the so-called lattice-gas automaton, from which later the lattice-Boltzmann method evolved.

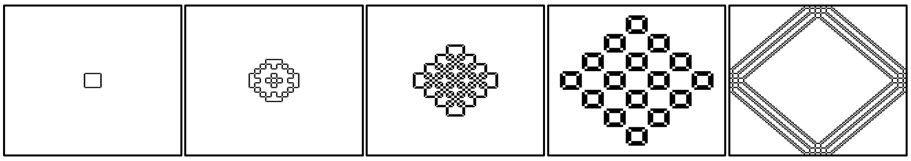

Figure 2. Evolution of a 256x256 cellular automaton.

2 Lattice Gas Automaton

The lattice gas is constructed as a simplified, fictitious molecular dynamic in which space, time, and the particle velocities are all discrete (Chen & Doolen, 1998). From this perspective, the lattice gas method is often referred to as a lattice gas cellular automaton. In general, a lattice gas cellular automaton consists of a regular lattice with particles residing on the nodes. A set of Boolean variables $n_i(\mathbf{x},t)\,(i=1,\cdots,M)$ describing the particle occupation is defined, where *M* is the number of directions of the particle velocity at each node. The evolution of the LG is as follows

$$n_i(\mathbf{x}+\mathbf{e}_i, t+1) = n_i(\mathbf{x},t) + \Omega_i(n(\mathbf{x},t)) \tag{2.1}$$

where $\mathbf{e}_i$ are the local particle velocities (again $i=1,\cdots,M$), and Ω_i is the collision operator that is a function of all particles $n_i\ (i=1,\cdots,M)$ involved in the collision. Starting from an initial state, the configuration of particles at each time step evolves in two sequential substeps: (a) *streaming* in which each particle moves to the nearest node in the direction of its velocity, and (b) *collision*, which occurs when particles arriving at a node interact and change their velocity directions according to scattering rules. For simplicity, the exclusion principle (not more than one particle being allowed at a given time and node with given velocity) is imposed for memory efficiency. This leads to a Fermi-Dirac local equilibrium distribution (Frisch et al., 1987).

3 Lattice-Boltzmann Method

The main feature of the lattice-Boltzmann method (LBM) is to replace the particle occupation variable n_i (Booleans) in Eq. (2.1) by single particle distribution functions (real variables) $f_i = \langle n_i \rangle$ and neglect individual particle motion and particle-particle correlations in the kinetic equations. The brackets $\langle\ \rangle$ denote an ensemble averaging. This procedure largely eliminates noise (which is present (and a problem) in lattice gases). In the LBM, the primitive variables are the averaged particle distributions, which are mesoscopic variables. Because the kinetic form is still the same as the lattice gas automaton, the advantage of locality in the kinetic approach is retained. The locality is essential to parallelism.

4 From Lattice-Boltzmann to Navier-Stokes

Since the lattice-Boltzmann method is a derivative of the lattice gas approach, the LBM will be introduced starting with a discrete kinetic equation for the particle distribution function, which is similar to the kinetic equation in lattice gas automata, see Eq. (2.1):

$$f_i(\mathbf{x}+\mathbf{e}_i\Delta x, t+\Delta t) = f_i(\mathbf{x},t) + \Omega_i(f(\mathbf{x},t)) \quad (i=0,1\cdots,M) \tag{4.1}$$

where f_i is the particle velocity distribution function along the *i*-th direction (please note that *i* now runs from 0 to *M*, in the LBM often a rest particle having zero-velocity and index 0 is introduced); $\Omega_i = \Omega_i\left(f\left(\mathbf{x},t\right)\right)$ is the collision operator that represents the rate of change of f_i as a result of the collision. The collision operator Ω_i depends on all M+1 particles (distribution functions) involved in the collision. The space and time increments are Δx and Δt respectively. If $\Delta x / \Delta t = \left|\mathbf{e}_i\right|$, Eqs. (2.1) and (4.1) have the same discretization. In the LBM, space is discretized in a way that is consistent with the kinetic equation, i.e. the coordinates of the neighbors of **x** are **x**+**e**$_i$.

The density ρ and momentum density $\rho\mathbf{u}$ are defined as moments of the distribution function f_i:

$$\rho = \sum_i f_i \qquad \rho\mathbf{u} = \sum_i f_i\mathbf{e}_i \tag{4.2}$$

with the sum over i=0...M. The collision operator Ω_i must satisfy mass and momentum conservation at each lattice:

$$\sum_i \Omega_i = 0 \qquad \sum_i \Omega_i\mathbf{e}_i = \mathbf{0} \tag{4.3}$$

If only the physics in the long-wavelength and low-frequency limit are of interest, the lattice spacing Δx and time step Δt in Eq. (4.1) can be regarded as small parameters of the same order ε. Performing a Taylor expansion in time and space, we obtain the following continuum form of the kinetic equation, accurate to second order in ε :

$$\frac{\partial f_i}{\partial t} + \mathbf{e}_i \cdot \nabla f_i + \varepsilon\left(\frac{1}{2}\mathbf{e}_i\mathbf{e}_i : \nabla\nabla f_i + \mathbf{e}_i \cdot \nabla\frac{\partial f_i}{\partial t} + \frac{1}{2}\frac{\partial^2 f_i}{\partial t^2}\right) = \frac{\Omega_i}{\varepsilon} \tag{4.4}$$

To derive the macroscopic hydrodynamic equations, we apply the Chapman-Enskog expansion, which essentially is a multi-scale expansion

$$\frac{\partial}{\partial t} = \varepsilon\frac{\partial}{\partial t_1} + \varepsilon^2\frac{\partial}{\partial t_2} \qquad \frac{\partial}{\partial x} = \varepsilon\frac{\partial}{\partial x_1} \tag{4.5}$$

The above assumes the diffusion time scale t_2 to be much slower than the convection time scale t_1. Likewise, the distribution function f_i can be expanded formally about the local equilibrium distribution function f_i^{eq}

$$f_i = f_i^{\mathrm{eq}} + \varepsilon f_i^{\mathrm{neq}} \tag{4.6}$$

Here f_i^{eq} depends on the local macroscopic variables (ρ and $\rho\mathbf{u}$) and should satisfy

$$\sum_i f_i^{\mathrm{eq}} = \rho \qquad \sum_i f_i^{\mathrm{eq}}\mathbf{e}_i = \rho\mathbf{u} \tag{4.7}$$

$f_i^{\text{neq}} = f_i^{(1)} + \varepsilon f_i^{(2)} + O(\varepsilon^2)$ is the non-equilibrium distribution function which should satisfy the following constraints

$$\sum_i f_i^{(k)} = 0 \quad \sum_i f_i^{(k)} \mathbf{e}_i = \mathbf{0} \quad k = 1,2 \tag{4.8}$$

Inserting f_i into the collision operator along with a Taylor expansion gives

$$\Omega_i(f) = \Omega_i(f^{\text{eq}}) + \varepsilon \frac{\partial \Omega_i(f^{\text{eq}})}{\partial f_j} f_j^{(1)} + \varepsilon^2 \left(\frac{\partial \Omega_i(f^{\text{eq}})}{\partial f_j} f_j^{(2)} + \frac{\partial^2 \Omega_i(f^{\text{eq}})}{\partial f_j \partial f_k} f_j^{(1)} f_k^{(1)} \right) + O(\varepsilon^3) \tag{4.9}$$

(summing of repeated indices is implied in this equation, and in the rest of the text). From Eq. (4.4) in the limit $\varepsilon \to 0$ (which implies $f_i = f_i^{\text{eq}}$) it follows $\Omega_i(f^{\text{eq}}) = 0$. This teaches us that we can linearize the collision operator:

$$\frac{\Omega_i(f)}{\varepsilon} = \frac{M_{ij}}{\varepsilon}\left(f_j - f_j^{\text{eq}}\right) \quad \text{with} \quad M_{ij} \equiv \frac{\partial \Omega_i(f^{\text{eq}})}{\partial f_j} \tag{4.10}$$

The matrix M_{ij} is the collision matrix, which determines the scattering rate between direction *i* and *j*. M_{ij} only depends on the angle between the directions *i* and *j*. Mass and momentum conservation imply:

$$\sum_{i=1}^{M} M_{ij} = 0 \quad \sum_{i=1}^{M} \mathbf{e}_i M_{ij} = \mathbf{0} \tag{4.11}$$

In the widely used lattice BGK (Bhatnagar-Gross-Krook, see Bhatnagar et al. 1954) collision operator, the distribution function relaxes to an equilibrium state at a single rate with time constant τ:

$$M_{ij} = -\frac{1}{\tau}\delta_{ij} \tag{4.12}$$

Then the collision term reads

$$\frac{\Omega_i}{\varepsilon} = -\frac{1}{\tau} f_i^{\text{neq}} = -\frac{1}{\tau}\left(f_i^{(1)} + \varepsilon f_i^{(2)}\right) \tag{4.13}$$

The lattice-BGK equation

$$f_i(\mathbf{x} + \mathbf{e}_i, t+1) = f_i(\mathbf{x}, t) - \frac{f_i - f_i^{\text{eq}}}{\tau} \tag{4.14}$$

forms the heart of many lattice-Boltzmann computer codes.

If we now substitute the LBGK collision operator, and the expansions of the distribution functions as given above in Eq. (4.4), equations to various orders of ε appear. For ε^0:

$$\frac{\partial f_i^{\text{eq}}}{\partial t_1} + \mathbf{e}_i \cdot \nabla_1 f_i^{\text{eq}} = -\frac{f_i^{(1)}}{\tau} \tag{4.15}$$

For order ε^1:

$$\frac{\partial f_i^{(1)}}{\partial t_1} + \frac{\partial f_i^{\text{eq}}}{\partial t_2} + \mathbf{e}_i \cdot \nabla f_i^{(1)} + \frac{1}{2}\mathbf{e}_i\mathbf{e}_i : \nabla\nabla f_i^{\text{eq}} + \mathbf{e}_i \cdot \nabla \frac{\partial f_i^{\text{eq}}}{\partial t_1} + \frac{1}{2}\frac{\partial^2}{\partial t_1^2} f_i^{\text{eq}} = \frac{f_i^{(2)}}{\tau} \tag{4.16}$$

Combining Eqs. (4.15) and (4.16) gives

$$\frac{\partial f_i^{(1)}}{\partial t_2} + \left(1 - \frac{2}{\tau}\right)\left(\frac{\partial f_i^{(1)}}{\partial t_1} + \mathbf{e}_i \cdot \nabla f_i^{(1)}\right) = -\frac{f_i^{(2)}}{\tau} \tag{4.17}$$

From Eqs. (4.15) and (4.17) the continuity equation and momentum balance can be derived:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho\mathbf{u} = 0 \tag{4.18}$$

$$\frac{\partial \rho\mathbf{u}}{\partial t} + \nabla \cdot \mathbf{\Pi} = \mathbf{0} \tag{4.19}$$

These equations are accurate to second order in ε. The momentum flux tensor $\mathbf{\Pi}$ has the form

$$\Pi_{\alpha\beta} = \sum_i e_{i\alpha} e_{i\beta} \left[f_i^{\text{eq}} + \left(1 - \frac{1}{2\tau}\right) f_i^{(1)} \right] \tag{4.20}$$

with $e_{i\alpha}$ the component of the velocity vector $\mathbf{e}_i$ in the α-th coordinate direction. Note that the momentum flux has an equilibrium part and a non-equilibrium part.

To specify the flux tensor, we need to specify the lattice structure and the equilibrium distribution. We consider a two-dimensional, square lattice. This relatively simple case has all the features that also apply to different lattices and number of dimensions. The set of velocity vectors can be written as

$$\mathbf{e}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \; \mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \; \mathbf{e}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \; \mathbf{e}_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \; \mathbf{e}_4 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \; \mathbf{e}_5 = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \; \mathbf{e}_6 = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \; \mathbf{e}_7 = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \; \mathbf{e}_8 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \tag{4.21}$$

The requirement for using the nine-velocity model (D2Q9 in the LB jargon: two-dimensions, nine speeds) instead of the simpler five-velocity model comes from considerations of lattice-symmetry. The lattice-Boltzmann equation cannot recover the correct Navier-Stokes equation unless sufficient lattice symmetry is present (Frisch et al 1986).

The Navier-Stokes equation has a second-order non-linearity. According to Chen et al (1992), the general form of the equilibrium distribution can be written up to $O(u^2)$:

$$f_i^{\mathrm{eq}} = \rho\left[a + b\mathbf{e}_i \cdot \mathbf{u} + c(\mathbf{e}_i \cdot \mathbf{u})^2 + du^2\right] \tag{4.22}$$

where a, b, c, and d are so-called lattice constants. This expansion of the distribution function only makes physical sense if the velocities are small compared to the (obviously finite) speed of sound of the lattice-Boltzmann system. Using the constraints as given in Eq. (4.7), the lattice constants can be obtained analytically and Eq. (4.22) can be written as

$$f_i^{\mathrm{eq}} = \rho w_i\left[1 + 3\mathbf{e}_i \cdot \mathbf{u} + \frac{9}{2}(\mathbf{e}_i \cdot \mathbf{u})^2 - \frac{3}{2}u^2\right] \tag{4.23}$$

The weight factors w_i are

$$w_0 = 4/9 \quad w_1 = w_3 = w_5 = w_7 = 1/9 \quad w_2 = w_4 = w_6 = w_8 = 1/36 \tag{4.24}$$

If we now get back to Eq. (4.20), the equilibrium part can be written as

$$\Pi_{\alpha\beta}^{eq} = \sum_i e_{i\alpha} e_{i\beta} f_i^{\mathrm{eq}} = p\delta_{\alpha\beta} + \rho u_\alpha u_\beta \tag{4.25}$$

$$\Pi_{\alpha\beta}^{(1)} = \left(1 - \frac{1}{2\tau}\right)\sum_i e_{i\alpha} e_{i\beta} f_i^{(1)} = \nu\left(\frac{\partial(\rho u_\beta)}{\partial x_\alpha} + \frac{\partial(\rho u_\alpha)}{\partial x_\beta}\right) \tag{4.26}$$

where $p=\rho/3$ is the pressure, and $\nu=(2\tau-1)/6$ is the kinematic viscosity. From the pressure relation the speed of sound c_s can be derived:

$$c_s^2 = \frac{\partial p}{\partial \rho} = \frac{1}{3} \tag{4.27}$$

The momentum balance then reads

$$\rho\left(\frac{\partial u_\alpha}{\partial t} + \frac{\partial u_\alpha u_\beta}{\partial x_\beta}\right) = -\frac{\partial p}{\partial x_\alpha} + \nu\frac{\partial}{\partial x_\beta}\left(\frac{\partial \rho u_\beta}{\partial x_\alpha} + \frac{\partial \rho u_\alpha}{\partial x_\beta}\right) \tag{4.28}$$

The momentum balance for a Newtonian fluid (the Navier-Stokes equation) for a compressible fluid reads

$$\rho\left(\frac{\partial u_\alpha}{\partial t}+\frac{\partial u_\alpha u_\beta}{\partial x_\beta}\right)=-\frac{\partial p}{\partial x_\alpha}+\frac{\partial}{\partial x_\beta}\left(\rho\nu\left(\frac{\partial u_\beta}{\partial x_\alpha}+\frac{\partial u_\alpha}{\partial x_\beta}-\frac{1}{3}\frac{\partial u_\gamma}{\partial x_\gamma}\delta_{\alpha\beta}\right)\right) \tag{4.29}$$

In the limit of constant density Eqs. (4.28) and (4.29) are the same. The limit of constant density can be effectuated in the LB scheme by keeping the fluid velocities well below the speed of sound (low Mach numbers), since $\delta\rho \propto \delta p / c_s^2 \propto \rho|\mathbf{u}|^2 / c_s^2$. That is, one should satisfy

$$\frac{|\mathbf{u}|^2}{c_s^2} \equiv \mathrm{Ma}^2 << 1 \tag{4.30}$$

5 Some Practical Aspects of the Lattice-Boltzmann Method for Single-Phase Flows

5.1 Implementation of the lattice-Boltzmann method in computer code

Implementing the above rules in computer code is fairly straightforward. This will be outlined here briefly in terms of Fortran-like pseudo code for a D2Q9 lattice-Boltzmann scheme (that scheme also was the subject of Section 4).

Define a main real array `f(n,i,j)`containing the distribution functions (or the LB fluid particles) at some specific moment in time. Experience learns that in most cases this can be a single-precision (`real*4`) array. The index `n` relates to the 19 velocity vectors (the numbering being the same as in Eq. (4.21)), and the indices `i,j` to the two coordinate directions that run from `1...nx`, and `1...ny` respectively (flow the domain is a rectangle with size `nx`·`ny`). The evolution of this system has two major steps: streaming and collision. In the streaming step, particles move to neighboring lattice sites. Computer code could look like this:

Code fragment

```
do j=1,ny
  do i=1,nx
    f(5,i,j)=f(5,i+1,j)
    f(7,i,j)=f(7,i,j+1)
    f(6,i,j)=f(6,i+1,j+1)
    f(8,i,j)=f(8,i-1,j+1)
  enddo
enddo

do j=ny,1,-1
  do i=nx,1,-1
```

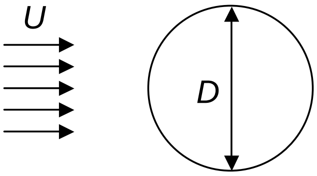

Figure 3. Schematic of the flow past a circular cylinder.

```
    f(1,i,j)=f(1,i-1,j)
    f(3,i,j)=f(3,i,j-1)
    f(2,i,j)=f(2,i-1,j-1)
    f(4,i,j)=f(4,i+1,j-1)
  enddo
enddo
```

end of code fragment

The above structure has been chosen such that we only need one field array `f`, i.e. we overwrite the array before the streaming step with the array after the streaming step.

The LBGK collision step, see Eq. (4.14) requires that we determine the equilibrium distribution According to Eq. (4.28) this needs computing the density and velocity per lattice node. For the latter we use Eq. (4.2).

It is (in my view) convenient to work in lattice-units while setting up an LB simulation. The unit of time is then set to 1, and is the time for an LB fluid particle (or distribution function) to travel to the neighboring lattice site. The unit of length is set to 1 as well and equals the spacing between two nearest-neighbor lattice nodes. The lattice spacing is uniform over the entire lattice and in the coordinate directions (square lattice in two dimensions, cubic in

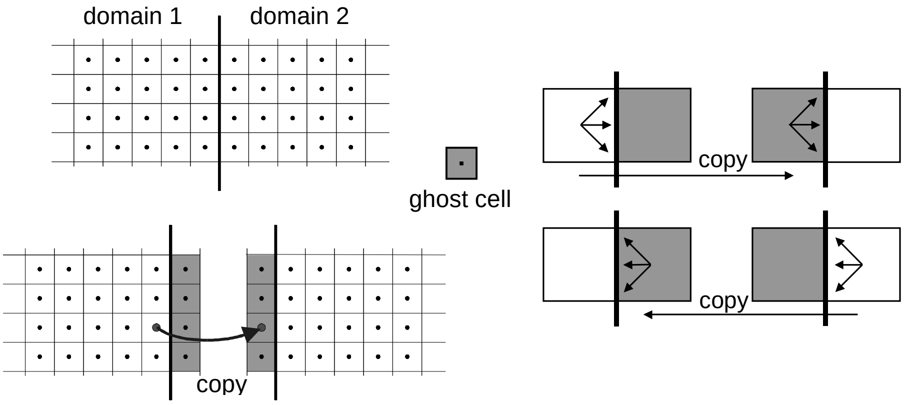

Figure 4. Parallellization through domain decomposition. Left: global view; right: at node level.

three directions) due to the coupling between velocity-space and physical space: particles have velocities such that they travel to their neighboring sites in exactly one time step.

Translating a flow case defined in physical units to LB units goes via dimensionless numbers. Suppose we would like to determine the flow past a square cylinder at flow and geometrical conditions as given in Figure 3. From Figure 3 a Reynolds number can be deduced: $\mathrm{Re} = UD/\nu$. Suppose we would like to simulate the case Re=100. In the LB simulation, we first choose a reasonable spatial resolution. Say we would like to resolve the flow such that the cylinder spans 10 lattice spacings. Then *D*=10 (in LB units). Flow velocities need to be such that Ma^2<<1 (Eq. (4.30)). This can be achieved by setting the free stream velocity *U* to 0.1 (LB units). For mimicking the physical case that has Re=100, the viscosity then needs to be set to ν=0.01 (again in LB units).

Parallellization of LB computer code is straightforward. The LB nodes only communicate with each other in the streaming step. For parallellization we decompose the flow domain in sub-domains. At the borders of the sub-domains we introduce ghost-cells. Before each streaming step the contents at the edges of each sub-domain is copied in the ghost cells of the neigboring domain (see Figure 4). Once this is done, the streaming step can be carried out in each subdomain according to the procedure presented earlier in this section.

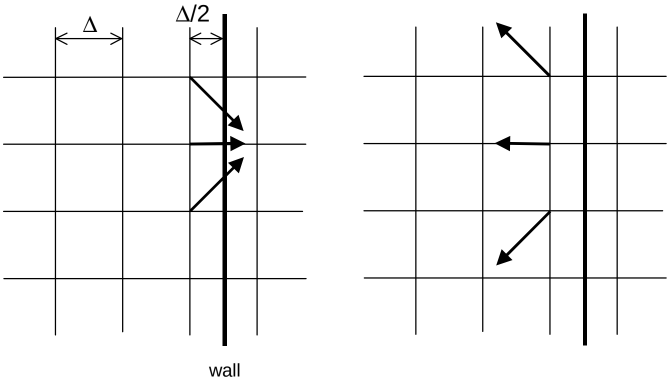

Figure 5. Streaming step for simulating a no-slip wall placed halfway lattice nodes.

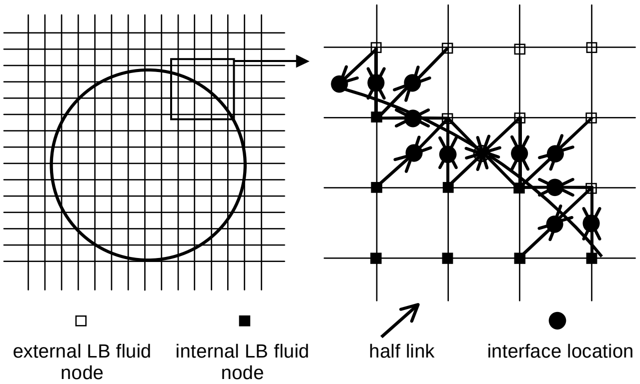

Figure 6. Definition of the halfway bounce-back rule for a circular particle. Right: zoomed-in view.

5.2 Setting boundary conditions

No-slip boundary conditions at solid walls can be set according the so-called bounce-back rule. In the half-way bounce back rule, the wall is located midway half a lattice-spacing away from a lattice node (see Figure 5). The particles propagating in the direction of the wall, bounce back at the wall and after exactly one time step arrive at the lattice node they left at the beginning of the time step. The half-way bounce-back method for no-slip walls is second order accurate (Rohde et al 2002), just as the discretization of the Navier-Stokes equation as given in Section 4. Placing the wall at a different location with respect to the grid reduces the accuracy at the boundary to first order.

If the no-slip wall has a non-zero velocity, the bounce-back method as described above can be generalized such that LB particles that bounce at the wall receive additional momentum due to the wall's motion. The half-way bounce-back rule including additional momentum from moving walls is the basis of Ladd's procedure (Ladd, 1994) of simulating solid-liquid suspensions with spherical particles. Figure 6 shows a representation of a circular object in Ladd's approach. The lattice-Boltzmann fluid particles bounce back halfway the link between two LB nodes, with the bounce back location close to the actual surface of the object. The circle (and sphere in 3D) are represented by a stair-step object. Such a schematic (rough) approach requires calibration. Usually a distinction is made between the input radius of a spherical particle, and the so-called hydrodynamic radius. The input radius relates to the particle as discretized in Figure 6. Of this particle the Stokes drag force is determined in a simulation of the sphere in a periodic domain. The result is compared to the analytical result (Hasimoto, 1959). Then the hydrodynamic radius is the radius of a sphere having a drag force equal to the one of the simulation, but now according to the analytical expression. Usually the hydrodynamic radius is slightly higher than the input radius (some 0.5 lattice spacing), it furthermore depends on the viscosity of the fluid.

Note that in this approach, there also is fluid inside the particle. This internal fluid has no physical meaning. It is there for computational convenience. The particles move relative to the grid and therefore cover and uncover LB nodes on a regular basis. If the particles have internal fluid, in the cover and uncover process fluid simply turns into internal and external fluid respectively and fluid mass is conserved easily. The consequence of having internal fluid is that the density of the particle always has to be at least the density of the fluid; the dynamics of lighter particles cannot be simulated. Also in setting up and solving the equations of motion of the particles, the internal fluid needs to be taken into account. This will be discussed in Section 6 of this chapter.

Aidun and co-workers (Aidun et al 1998, Ding and Aidun 2003) have adapted Ladd's method such that internal fluid is largely avoided. The price is the execution of a dedicated procedure to conserve fluid mass as good as possible.

In recent years, immersed boundary methods are gaining popularity in mainstream CFD based on e.g. the finite volume method, or spectral methods (Goldstein et al 1993). Similar techniques can be employed in an LB context. The idea is to apply body forces on the fluid such that at prescribed locations the fluid has a prescribed velocity (equal to the velocity of a solid wall). An advantage of the method is that the locations can (in principle) be chosen independent of the grid. In the case of defining a moving sphere, the forcing method works as follows: The sphere's surface is defined as a set of M control points $\mathbf{r}_j^{(n)}$ (j=1…M) on their surface, where the superscript (n) now indicates the moment in time. There is no restriction on the position of these points in the flow domain; they do not need to coincide with lattice sites. At these points we require a velocity equal to $\mathbf{w}_j^{(n)}= \mathbf{v}_\mathrm{p}+\mathbf{\Omega}_\mathrm{p}\times(\mathbf{r}_j^{(n)}-\mathbf{r}_\mathrm{p}^{(\mathrm{n})})$, with $\mathbf{v}_\mathrm{p}$ the linear velocity and $\mathbf{\Omega}_\mathrm{p}$ the angular velocity of the particle.

The above demand can be achieved effectively through a control algorithm (Goldstein et al 1993) that at each time step determines the (interpolated) mismatch between the actual flow velocity and the prescribed flow velocity at the control points, and then adapts the force field in such a way that it suppresses the mismatch. The deviation between the actual and the prescribed velocity ($\mathbf{d}_j^{(n)}$) is determined by a second-order interpolation of the flow velocities at the lattice sites:

$$\mathbf{d}_j^{(n)} = \mathbf{w}_j^{(n)} - \sum_k G_k \left(\mathbf{r}_j^{(n)}\right)\mathbf{u}_k^{(n)} \tag{5.1}$$

where the sum is over the lattice sites in the vicinity of $\mathbf{r}_j^{(n)}$, $\mathbf{u}_k$ is the fluid velocity at lattice site k, and G_k are the interpolation coefficients. These coefficients also serve to distribute the forces that reduce the deviation $\mathbf{d}_j^{(n)}$, over the lattice sites:

$$\mathbf{f}_k^{(n)} = \mathbf{f}_k^{(n-1)} + q\sum_j G_k \left(\mathbf{r}_j^{(n)}\right)\mathbf{d}_j^{(n)} \tag{5.2}$$

with $\mathbf{f}_k^{(n)}$ the force acting on the fluid at lattice node k and moment n, and q a relaxation factor. Also the forcing method needs calibration of the hydrodynamic diameter. For this we use the same procedure as proposed by Ladd.

Other sorts of boundary conditions can be achieved in the LBM relatively easily and along intuitive lines:

- Zero-shear at flat walls is achieved through specular reflection of the LB particles at the wall.
- Zero gradient conditions (at inflows and outflows) are achieved by copying distribution functions in the direction normal to the boundary; at inflows the forcing scheme (see above) can then be used to impose velocity profiles.
- Periodic boundaries imply copying outgoing distribution functions to the other side of the domain where they enter again.

5.3 Alternative collision operators

The LBGK collision operator, based on a single time constant with which distribution functions relax towards equilibrium gets unstable for low viscosity values. This has led to research towards more stable schemes. Two more stable schemes are mentioned here: the multiple relaxation time (MRT) scheme (Lallemand and Luo, 2000), and a scheme due to Somers (1993).

For a MRT-LB model with M velocities, a set of velocity distribution functions $f_i \quad \left(i = 0,1\cdots,M\right)$ is defined. The collision (and this is different from LBGK) is executed in moment space (not in velocity space). Moment space and velocity space are connected via a linear transformation

$$m_i = T_{ij} f_j \qquad f_i = \left(T^{-1}\right)_{ij} m_j \tag{5.3}$$

The evolution equation then reads

$$f_i\left(\mathbf{x} + \mathbf{e}_i \Delta x, t + \Delta t\right) = f_i\left(\mathbf{x}, t\right) + \Omega_i\left(f\left(\mathbf{x}, t\right)\right) = -\left(T^{-1}\right)_{ij} \Lambda_j \left(m_j - m_j^{eq}\right) \tag{5.4}$$

The moments are related to density, momentum, strain, and energy. Their equilibriums are functions of the conserved quantities which are mass and momentum. The *M*+1 coefficients Λ_j determine the viscosity (as did the relaxation time) and are used to enhance the stability of the scheme.

The scheme due to Somers has been described in detail by Eggels and Somers (1995). It goes along similar lines as the MRT approach, i.e. its collision operator acts on moments of the velocity distribution function. Furthermore, it uses a staggered discretization in space and time.

6 Direct Numerical Simulations of Solid-Liquid Suspensions

In this section we will discuss a methodology for directly simulating solid-liquid suspensions. In these simulations we resolve the solid-liquid interface and the flow of the interstitial fluid (the latter we do with the LBM), i.e. the spherical solid particles have finite size and the flow around the particles is directly simulated. The forcing method described in Section 5.2 is used for setting the no-slip boundary condition at the sphere's surfaces. At the end of this section we will briefly discuss two examples of such simulations: a turbulent suspension, and solid-liquid fluidization.

Our starting point is a fully periodic, three-dimensional domain containing fluid and solid particles with a spherical shape. In the fluidization cases, the flow is driven by a net gravity force acting on the particles, and a pressure gradient acting on the fluid to balance gravity. In the case of the turbulent suspension, a random body force that can generate turbulence with prescribed properties agitates the suspension.

The fluid flow and the particle motion are coupled by demanding that at the surface of the sphere the fluid velocity matches the local velocity of the solid surface (that is the sum of the linear velocity $\mathbf{v}_\mathrm{p}$ and $\mathbf{\Omega}_\mathrm{p}\times(\mathbf{r}-\mathbf{r}_\mathrm{p})$ with $\mathbf{\Omega}_\mathrm{p}$ being the angular velocity of the particle); in the forcing scheme this is accomplished by imposing additional forces on the fluid at the surface of the solid sphere (which is then distributed to the lattice nodes in the vicinity of the particle surface). The collection of forces acting on the fluid at the sphere's surface and its interior is subsequently used to determine the hydrodynamic force and torque acting on the sphere (action = –reaction).

The effective body force on the fluid (in the fluidization case) mentioned above can be related to the gravitational acceleration, $\mathbf{g} = -g\mathbf{e}_\mathrm{z}$, as follows. The net gravity force acting on each spherical particle is $\mathbf{F}_\mathrm{G} = -\left(\rho_\mathrm{s} - \overline{\rho}\right)\frac{\pi}{6} d_\mathrm{p}^3 g\mathbf{e}_\mathrm{z}$, and the force per unit volume acting on the (internal and external) fluid is

$$\mathbf{f}_{\mathrm{B}} = (\bar{\rho} - \rho_{\mathrm{f}}) g \mathbf{e}_{\mathrm{z}} \tag{6.1}$$

with $\bar{\rho} = \bar{\phi}\rho_{\mathrm{s}} + (1-\bar{\phi})\rho_{\mathrm{f}}$ the density of the fluid-solid mixture, and $\bar{\phi}$ the overall (spatially averaged) solids volume fraction in the periodic domain.

The fluid inside the spherical particles is an artefact of the forcing scheme. As long as the density of the solid is higher than the density of the fluid, the effects of the internal fluid can be effectively corrected for: The force $\mathbf{F}_{\mathrm{LB}}$ acting on the fluid determined by the forcing method is the sum of the force needed to accelerate the internal fluid and the force of the particle acting on the external fluid. Since the internal fluid largely behaves as a solid body, one can partition $\mathbf{F}_{\mathrm{LB}}$ as follows: $\mathbf{F}_{\mathrm{LB}} = \mathbf{F}_{\mathrm{LB,1}} + \mathbf{F}_{\mathrm{ext}}$ where the force $\mathbf{F}_{\mathrm{LB,1}}$ is the component that ensures that the internal fluid translates with the particle; $\mathbf{F}_{\mathrm{ext}}$ is the force on the external fluid due to the particle. The overall linear momentum balance for the internal fluid can be written as

$$\rho_{\mathrm{f}} \frac{\pi}{6} d_{\mathrm{p}}^3 \frac{\mathrm{d}\mathbf{v}_{\mathbf{p}}}{\mathrm{d}t} = +\mathbf{F}_{\mathrm{LB,1}} + (\bar{\rho} - \rho_{\mathrm{f}}) \frac{\pi}{6} d_{\mathrm{p}}^3 g \mathbf{e}_{\mathbf{z}} = \mathbf{F}_{\mathrm{LB}} - \mathbf{F}_{\mathrm{ext}} + (\bar{\rho} - \rho_{\mathrm{f}}) \frac{\pi}{6} d_{\mathrm{p}}^3 g \mathbf{e}_{\mathbf{z}} \tag{6.2}$$

where it has been recognized that the internal fluid translates with the particle. The corresponding equation for the particle is then

$$\rho_{\mathrm{s}} \frac{\pi}{6} d_{\mathrm{p}}^3 \frac{\mathrm{d}\mathbf{v}_{\mathrm{p}}}{\mathrm{d}t} = -\mathbf{F}_{\mathrm{ext}} + (\rho_{\mathrm{s}} - \bar{\rho}) \frac{\pi}{6} d_{\mathrm{p}}^3 g \mathbf{e}_{\mathbf{z}} \tag{6.3}$$

Lubrication forces which arise because of inadequate resolution of the flow in between neighbouring particles and those arising from direct particle-particle interactions (e.g., collision) will be added to the right hand side later. Combining Eqs. (6.2) and (6.3), we get

$$(\rho_{\mathrm{s}} - \rho_{\mathrm{f}}) \frac{\pi}{6} d_{\mathrm{p}}^3 \frac{\mathrm{d}\mathbf{v}_{\mathrm{p}}}{\mathrm{d}t} = -\mathbf{F}_{\mathrm{LB}} - (\rho_{\mathrm{s}} - \rho_{\mathrm{f}}) \frac{\pi}{6} d_{\mathrm{p}}^3 g \mathbf{e}_{\mathbf{z}} \tag{6.4}$$

Following the same reasoning, we obtain the following angular momentum balance:

$$(\rho_{\mathrm{s}} - \rho_{\mathrm{f}}) \frac{\pi}{60} d_{\mathrm{p}}^5 \frac{\mathrm{d}\mathbf{\Omega}_{\mathbf{p}}}{\mathrm{d}t} = -\mathbf{T}_{\mathrm{LB}} \tag{6.5}$$

with $\mathbf{T}_{\mathrm{LB}}$ the torque as determined by the forcing method to impose the no-slip conditions at the sphere's surface.

In order to test if the above procedure represents the dynamics of spheres immersed in liquid properly, we considered the transient motion of a single sphere that is accelerated from rest under the influence of gravity. In the limit of zero Reynolds number in an unbounded fluid the equation of motion of the sphere has been derived by Maxey and Riley (1983). Results obtained by integrating the Maxey and Riley equation showed excellent agreement with the LB simulations combined with the above equations of motion. Ten Cate et al. (2002) compared the results on sedimentation of a single sphere in a limited-size container at higher Reynolds numbers (up to Re = 30) obtained through lattice-Boltzmann simulations and forcing boundary conditions with particle image velocimetry (PIV) data and found good agreement in terms of the sphere's trajectory and the fluid flow field around the moving sphere.

We also take into consideration the interaction between particles through binary, hard-sphere collisions and lubrication forces. For the former, we apply an event-driven collision algorithm: we move the collection of particles until two particles get into contact. At that moment we carry out the collision (i.e. update the velocities of the two particles taking part in the collision). Subsequently, the movements of all particles are continued until the next collision or until the end of a LB time step. The collision model that we apply (described in detail in Yamamoto et al., 2001) has two parameters: a restitution coefficient *e* and a friction coefficient μ. As the default situation we consider fully elastic, frictionless collisions (e = 1, μ = 0).

When two particles are at close proximity, with their separation being of the order of or less than the lattice spacing, the hydrodynamic interaction between them will not be properly resolved in the LB simulations. Therefore, we explicitly impose lubrication forces on the particles, in addition to the hydrodynamic forces stemming from the LBM. The general framework for lubrication forces and torques acting on two particles (1 and 2) as a result of the relative motion of their surfaces can be written in the form of the following vector equation:

$$\begin{bmatrix} \mathbf{F}_{\mathrm{lub},1} \\ \mathbf{T}_{\mathrm{lub},1} \\ \mathbf{T}_{\mathrm{lub},2} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11} & -\mathbf{B}_{11} & \mathbf{B}_{22} \\ \mathbf{B}_{11} & \mathbf{C}_{11} & \mathbf{C}_{12} \\ -\mathbf{B}_{22} & \mathbf{C}_{12} & \mathbf{C}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{v}_{\mathrm{p},12} \\ \mathbf{\Omega}_{1} \\ \mathbf{\Omega}_{2} \end{bmatrix} \tag{6.6}$$

with $\mathbf{F}_{\mathrm{lub},2} = -\mathbf{F}_{\mathrm{lub},1}$ and $\mathbf{v}_{\mathrm{p},12} = \mathbf{v}_{\mathrm{p},1} - \mathbf{v}_{\mathrm{p},2}$ (Kim and Karilla, 1991; Nguyen and Ladd, 2002). In the tensors $\mathbf{A}_{11}$, $\mathbf{B}_{11}$, $\mathbf{B}_{22}$, $\mathbf{C}_{11}$, $\mathbf{C}_{22}$, and $\mathbf{C}_{12}$, we only use the leading order terms in the parameter $\frac{d_{\mathrm{p}}}{h}$, with *h* the minimum spacing of the particle surfaces. For the radial lubrication force (contained in the diagonal elements of the $\mathbf{A_{11}}$ matrix in the equation $\mathbf{F}_{\mathrm{lub},1} = \mathbf{A}_{11}\,\mathbf{v}_{\mathrm{p},12}$,) the leading order is $\frac{d_{\mathrm{p}}}{h}$, while for the tangential lubrication forces and torques it is $\ln\left(\frac{d_{\mathrm{p}}}{h}\right)$ (Kim and Karilla, 1991). Two modifications to the above expressions were implemented to tailor them to our numerical needs:

(a) Lubrication only acts if particle separation is less than $\delta = 0.1 d_{\mathrm{p}}$ (which is equivalent to roughly one lattice spacing in the default resolutions we use). To smoothly switch on/off the lubrication force at $h = \delta$, in the lubrication expressions $\frac{d_\mathrm{p}}{h}$ is replaced by $\frac{d_\mathrm{p}}{h} - \frac{d_\mathrm{p}}{\delta}$, and $\ln\left(\frac{d_\mathrm{p}}{h}\right)$ by $\ln\left(\frac{d_\mathrm{p}}{h}\right) - \ln\left(\frac{d_\mathrm{p}}{\delta}\right)$ (Nguyen and Ladd, 2002).

(b) The lubrication force saturates once the particles are very close (at$10^{-4}d_\mathrm{p}$). The latter restriction we use for numerical reasons (to avoid high force levels and associated instabilities) but also with the surface roughness of the particles and/or the mean-free-path of the fluid in mind.

The time-step-driven (LBM) and the event-driven (collisional) parts of the simulation have been combined by first performing the LBM time step from t to $t+\Delta t$ and subsequently moving the particles until also the particle system has advanced Δt in time. Since in dense systems usually more than one collision occurs during Δt, the particle motion algorithm sets a number of sub-time-steps, the number being equal to one plus the number of (potential, see below) collisions.

At the start of every particle motion sub-time-step, we update the lubrication forces and torques. Then we move the particles over the sub-time-step, i.e. until the next potential collision (or until $t+\Delta t$ is reached). At the new positions of the particles we again determine the lubrication force and torque. The linear and angular velocities of the particles are now updated according to the average of the lubrication forces and torques at the beginning and at the end of the sub-time-step. The velocity update may result in the collision not to occur: in that situation the lubrication forces were sufficiently strong to change the sign of the relative particle velocity so that a hard-sphere collision was prevented.

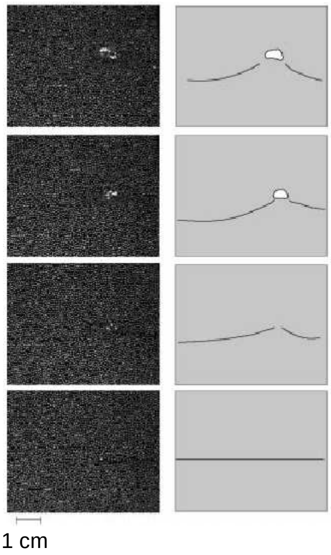

Figure 7. Experiment showing the onset of a bubble in a flat fluidized bed. Left: snapshots of the bed; right: schematization of the observations. Reprinted from Duru and Guazelli (2002).

6.1 Some results for solid-liquid fluidization

Two-dimensional waves and the onset of bubbles have been studied experimentally by Duru and Guazzelli (2002). They used flat, liquid-fluidized beds that could not develop three-dimensional structures, and allowed for good visible observations, well-resolved void fraction measurements, and particle tracking velocimetry measurements. Figure 7 shows a typical result of their experiments: the development of a bubble-like void, starting from a planar wave instability. In this case, steel beads (density $7.8{\cdot}10^3$ kg/m^3) with a diameter of d_p = 1 mm were fluidized with water in a domain that was $120d_\mathrm{p}$ wide, $12d_\mathrm{p}$ thick and some $2000d_\mathrm{p}$ high.

Such two-dimensional structures can indeed be simulated using the approach described above. In order to see the evolution of two-dimensional structures, we have performed simulations in a $24d_\mathrm{p}\times6d_\mathrm{p}\times20d_\mathrm{p}$ periodic domain (the $20d_\mathrm{p}$ being in the streamwise direction). As an initial condition for the particle positions and velocities (translational and rotational), we juxtaposed four copies of a fully developed planar wave [computed in a $6d_\mathrm{p}\times6d_\mathrm{p}\times20d_\mathrm{p}$ periodic domain]. This simulation was performed at a lower resolution such that the hydrodynamic diameter of the particles was set to be 12 lattice units. The density ratio was set to 8. The viscosity and body force were chosen such that the terminal velocity of a single bead was 0.04 (in lattice units) and the Reynolds number based on the terminal velocity matched the value in the experiments (Re = 400). The collisions were smooth and elastic (μ = 0, e = 1).

In Figure 8 we show how the numerical system developed a bubble very similar to the experimental observations: the initially plane wave buckles and at its crest forms a bubble-like void. The particle velocity field in the vicinity shows qualitative similarity with the field measured by Duru and Guazzelli (2002); see Figure 9. We again note that non-ideality of bead-bead collisions is not an essential condition for resolving the behavior of void fraction

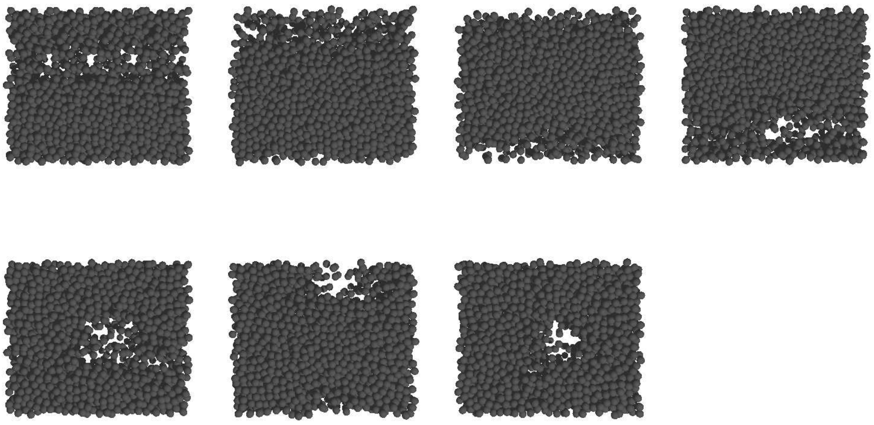

Figure 8. Series of snapshots showing bubble formation. The upper left frame shows the initial condition. The time-spacing between the subsequent (left-to-right, top-to-bottom) frames is

$$0.057\frac{d_p^2}{\nu}.$$

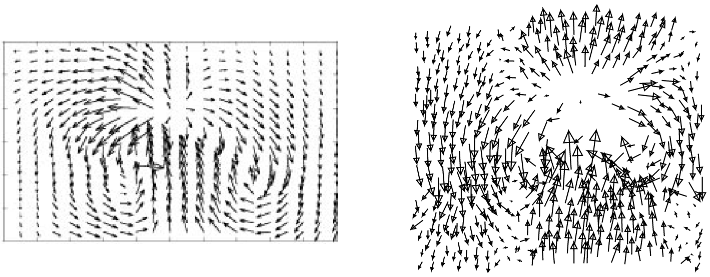

Figure 9. Measured (left, reprinted from Duru and Guazelli 2002), and simulated particle velocities in the vicinity of a bubble.

instabilities in fluidized beds. The domain size in the flow direction is too short for the bubble to behave as an isolated bubble. The periodic system resembles a bubble train, which generally has a higher velocity than an isolated bubble. If we estimate the bubble rise velocity and translate it back to the experimental steel-water system, the bubble rise velocity in the simulation is approximately 15 cm/s. The bubble radius is approximately 0.4 cm. This bubble size and rise velocity combination is at the lower end of the range of rise velocities for bubble trains which is 15 to 22 cm/s according to the experiments (Duru and Guazzelli, 2002).

6.2 DNS of turbulently agitated solid-liquid suspensions

Another example of the application of our approach for directly simulating solid-liquid suspensions is in the field of turbulence and has been reported earlier by ten Cate et al. (2004). We again define a three-dimensional system of solid particles dispersed in a liquid

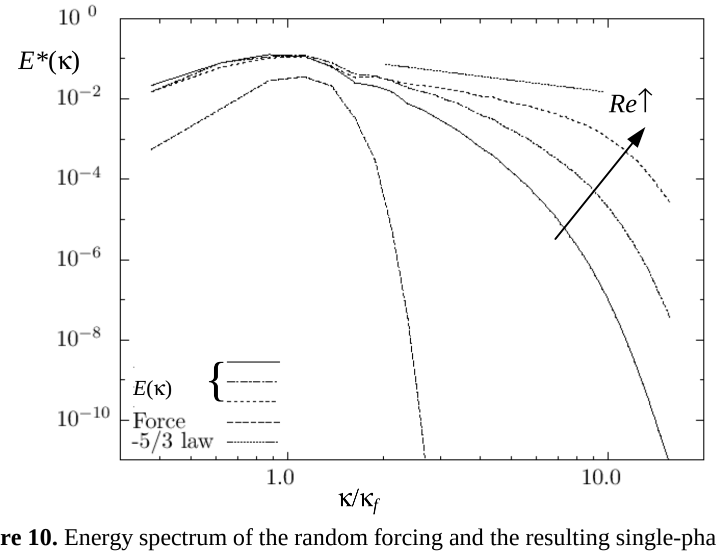

Figure 10. Energy spectrum of the random forcing and the resulting single-phase fluid flow. The wavenumber κ was normalized with the peak-forcing wavenumber κ_f. The energy E is non-dimenionalized with $u_*^2 l_*$ with u_* the rms velocity, and $l_*=2\pi/\kappa_f$.

with fully periodic boundary conditions. Instead of driving the system by a uniform body force on the particles and a (opposing) body force on the fluid (mimicking the pressure gradient), we now force the system in a random manner as to generate homogeneous, isotropic turbulence. The procedure for doing this is due to Alvelius (1999); Ten Cate et al (2006) adapted the method so that it could be combined with the lattice-Boltzmann method. In Figure 10 we show results for single-phase turbulence in terms of the energy spectrum. The spectrum of the random forcing is limited to small wavenumbers (large scales). With increasing Reynolds number, the flow develops smaller and smaller scales (the spectrum extends to higher wavenumbers), and develops and inertial subrange characterized by a -5/3 slope in the energy spectrum.

The presence of particles changes the spectrum (see Figure 11): the particles create turbulence at scales comparable to and smaller than the particle diameter. One of our interests in this study was the way particles collide (collision frequencies and intensities). In this respect it was interesting to study the PDF of the time between two collisions of a particle (as given in Figure 12). For "long times" this PDF is exponential indicating Poisson statistics (with a steeper slope for more dense systems). These collisions are uncorrelated events. For "short times" the PDF deviates from exponential and shows a peak towards zero time. These are correlated events: once turbulence has brought two (or more) particles in each others vicinity they tend to cluster due to short range hydrodynamic interaction (lubrication) and undergo many (weak) collisions at short time intervals. Eventually the particles in the cluster are separated when a strong enough eddy comes by.

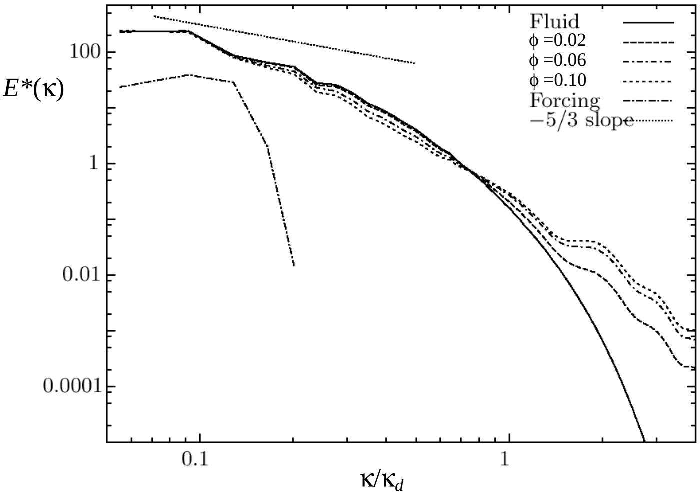

Figure 11. Energy spectra of the two-phase simulations (with solids volume fractions ϕ) compared to the single-phase (fluid) spectrum. The wavenumber κ was normalized with the particle size wavenumber $\kappa_d=2\pi/d_p$. The energy E is non-dimenionalized with $\varepsilon^{2/3}\eta^{5/3}$ with ε the average energy dissipation rate, and η the Kolmogorov lenght scale.

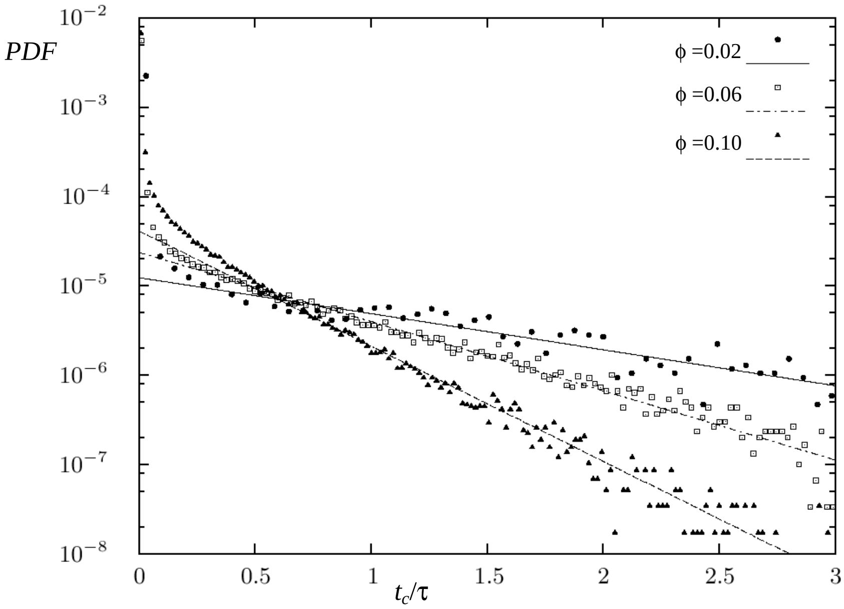

Figure 12. PDF of the time between two collisions of a particle for three solids volume fractions. The lines are linear fits of the tails of the distributions. The collision time has been made dimensionless with the Kolmogorov time scale τ.

7 Single Phase Turbulence

A turbulent flow exhibits an irregular behavior in space and time. A typical example of a time signal corresponding to a turbulent flow quantity is shown in Figure 13 where the streamwise velocity recorded in a turbulent pipe flow is shown as a function of time. At first glance, the velocity may seem to behave randomly. Detailed studies, however, have shown that turbulent flows are not completely random in space and time. They contain spatial (coherent) structures that evolve in time. These structures are often referred to as eddies, as they are usually associated with rotating motions of fluid flow. A fundamental result of turbulence theory is that these eddies are not all of a particular size, but that an often broad continuous range of large to small eddies exists. If we return to Figure 13, and carefully study the temporal evolution of the turbulence signal shown there, we see that in this signal both “fast” and “slow” temporal variations occur, that might be associated to small and large eddies respectively. In general, the size of the largest eddies in a turbulent flow is determined by the geometry of the flow configuration. Here it is characterized by a length scale l. Typical values of l for wall-bounded, shear-driven turbulence are $l \approx 0.1L$ with L a length scale corresponding to the flow geometry (e.g. the pipe diameter). Besides a length scale, these large eddies also have a velocity scale denoted by u. From this we can deduce that large-scale eddies have a typical time-scale proportional to l/u, and a turbulent kinetic energy (per

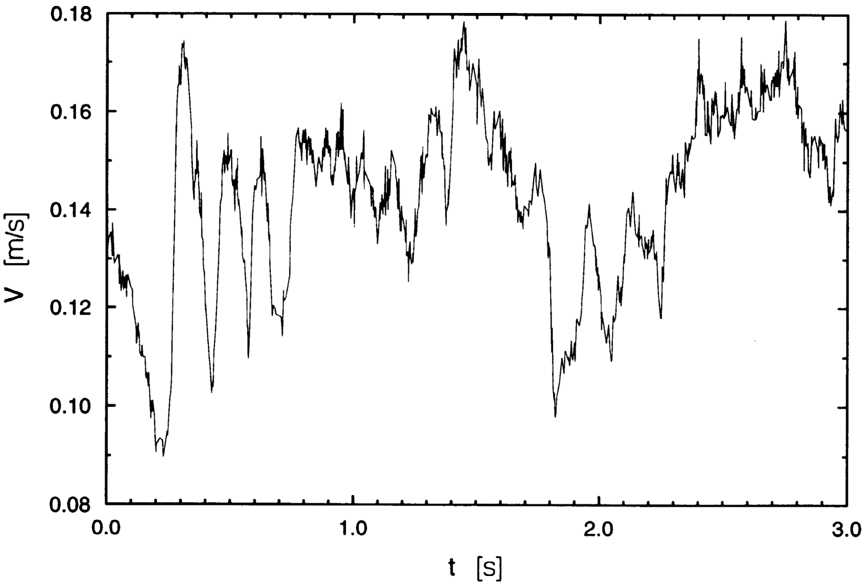

Figure 13. Evolution of the streamwise velocity as a function of time measured in fully developed turbulent pipe flow (reprinted from Eggels, 1994; courtesy J. den Toonder).

unit mass) proportional to u^2. This kinetic energy is extracted from the mean flow by interaction of the mean flow and the turbulent fluctuations.

The smaller eddies do not extract their kinetic energy directly from the mean flow but are fed by a continuous decay of (unstable) large eddies which break up into smaller ones. These smaller ones in turn decay to even smaller eddies until this cascade reaches the smallest scales of turbulent motion (the energy cascade). The length and velocity scales of these smallest eddies are determined by the amount of kinetic energy that is being transformed along the energy cascade from large to small eddies, and by the molecular viscosity of the fluid that eventually (at the smallest dynamical scales) dissipates the energy.

The loss of kinetic energy of the large-scale eddies is represented by the dissipation rate ε (energy per unit time and unit mass). In turbulence theory, it is assumed that the dissipation rate is independent of the turbulent micro-structure (or the small eddies) since it is fully determined by what happens at the large scales. This is expressed by the following:

$$\varepsilon = \frac{u^3}{l} \tag{7.1}$$

This relation can be interpreted as the kinetic energy of the macro-structure eddies (u^2) being transferred to smaller scales (break-up of bigger eddies into smaller eddies) during their lifetime (l/u).

The smallest eddies are fully controlled by the energy transfer (or, equivalently, by the dissipation rate ε), and by the molecular viscosity ν, they are e.g. decoupled from the flow geometry. By means of dimensional analysis, the length, velocity, and time scales of these smallest eddies can be determined:

$$\eta = \left(\frac{\nu^3}{\varepsilon} \right)^{1/4} \qquad \upsilon = (\nu\varepsilon)^{1/4} \qquad \tau = \left(\frac{\nu}{\varepsilon} \right)^{1/2} \tag{7.2}$$

These are the celebrated Kolmogorov scales. Since the dissipation rate is known in terms of macro-structure properties, we can easily deduce relations between the various scales of macro-structure and micro-structure. Substituting Eq. (7.1) into Eq. (7.2) yields

$$\frac{\eta}{l} = \mathrm{Re}_l^{3/4} \qquad \frac{\upsilon}{u} = \mathrm{Re}_l^{-1/4} \qquad \frac{\tau}{l/u} = \mathrm{Re}_l^{-1/2} \qquad \text{with} \quad \mathrm{Re}_l = \frac{ul}{\nu} \tag{7.3}$$

For large Reynolds numbers Re$_l$, the scales of the micro-structure become much smaller compared to those of the macro-structure. In other words, the energy cascade process determines the scales of the micro-structure in such a way that the smallest eddies can transform their kinetic energy into internal energy (heat) by means of molecular viscosity. If e.g. the kinetic energy of the macro-structure is increased (i.e. if Re$_l$ becomes larger), then the scales of the micro-structure become smaller (compared to *l* and *u*) in order to more effectively transform the increased amount of kinetic energy into internal energy.

The appearance of a broad range of scales in a turbulent flow with the macro-structure characterized by its large eddies on one hand and the micro-structure with its small-scale eddies on the other, is our point of departure to illustrate the principles of the simulation techniques described here.

8 Numerical Simulation of Fluid Flow

For an incompressible, Newtonian fluid, the conservation of mass and momentum are as follows

$$\nabla \cdot \mathbf{v} = 0 \tag{8.1}$$

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{v} \tag{8.2}$$

with **v** the velocity vector, ρ the fluid density, and p the pressure. The Reynolds number (introduced in Section 7), can be interpreted as the ratio of the advection term $\mathbf{v} \cdot \nabla \mathbf{v}$ and the viscous term $\nu \nabla^2 \mathbf{v}$ both scaled by the macro-scales *u* and *l*:

$$\frac{|\mathbf{v} \cdot \nabla \mathbf{v}|}{|\nu \nabla^2 \mathbf{v}|} \propto \frac{u \frac{1}{l} u}{\nu \frac{1}{l^2} u} = \frac{ul}{\nu} = \mathrm{Re}_l \tag{8.3}$$

For large Re_l, the nonlinear advection term dominates over the viscous term and (in general) the flow will be turbulent. The nonlinearity of the advection term is the reason for the appearance of a broad range of scales in turbulent flows. If the viscous term dominates over the advection term (small Re_l), the flow is laminar and has a regular flow pattern.

Since the equations which describe the flow field are known, it should be possible (in principle) to solve them in a discretized form by means of a computer. Such a numerical simulation should resolve the spatial and temporal evolution of the flow field in all detail to capture all relevant flow phenomena. For laminar flow with low Re_l, the equations of motion can be discretized and solved straightforwardly from a computational point of view. For turbulent flows, the situation is different and more complicated.

8.1 Direct numerical simulation (DNS)

Discretized versions of the partial differential equations involve a grid (spatial discretization) and time steps. Let the distance between two sequential point in space and time be denoted by Δx and Δt respectively. Turbulent flows are characterized by a broad range of length and time scales that should be resolved in all detail by the simulation. Hence, Δx and Δt should be proportional to the smallest length and time scales respectively: $\Delta x \propto \eta$ and $\Delta t \propto \tau$. A more restrictive criterion, however, must sometimes be applied for Δt in which $\Delta t \propto \tau_\mathrm{a}$, with τ_a the time scale associated with a small-scale eddy passing a fixed point when being advected by the macro-structure velocity, i.e. $\tau_\mathrm{a} \propto \eta / u$.

In Eq. (7.3) we related the macro and micro-structure. Using $l \approx 0.1L$:

$$N_L = \frac{L}{\Delta x} \propto \frac{10l}{\eta} = 10\,\mathrm{Re}_l^{3/4} \tag{8.4}$$

For the ratio T and Δt with T a time interval during which the flow field is monitored, a similar expression can be obtained. As for the length l and L, we have to relate T to the time-scale of the macro-structure l/u. To obtain a correct impression of the turbulent flow, it should be at least monitored for several time scales l/u, say 50 times. Then we find

$$N_T = \frac{T}{\Delta t} \propto \frac{50 l / u}{\tau_\mathrm{a}} = 50\,\mathrm{Re}_l^{3/4} \tag{8.5}$$

in which we used the (in general) most restrictive criterion to the time step.

The computational time T_C for direct numerical simulations scales at least as the total number of grid points (being N_L^3) and the number of time steps N_T. Memory requirements M_C scale with N_L^3. As a result

$$T_\mathrm{C} \propto \mathrm{Re}_l^3 \quad M_\mathrm{C} \propto \mathrm{Re}_l^{9/4} \tag{8.6}$$

(for wall bounded flows with boundary layers the above scalings are even on the low side since generally more resolution is required close to walls). Both M_C and T_C are proportional to Re_l raised to a positive power larger than 1. With increasing Re_l, the computational effort rapidly increases. It is obvious that for strongly turbulent flows (say $\mathrm{Re}_l=10^4$) DNS cannot be performed due to the limited capacity of present-day (super)computers. For relatively modest Re_l, DNS is a very useful technique for studying turbulence in all its detail.

8.2 Large-eddy simulation (LES)

A remedy to overcome the limitations of DNS with increasing Re_l is to reduce the range of scales that are resolved on the numerical grid. A viable concept is to remove the small-scale eddies by a spatial filtering procedure, and to resolve the large-scale eddies only. This approach is called large-eddy simulation (LES). The separation of large and small scales is inspired by observations that the large eddies of the macro-structure are mostly anisotropic. Furthermore, they depend on the geometry of the flow considered. On the other hand, the small eddies of the micro-structure can be considered to be closer to isotropic. They are much less dependent on the flow geometry, as they are fed by the energy cascade in which the geometry information present in the larger eddies gets lost due to break-ups. Therefore, the micro-structure-eddies may tentatively be regarded more or less universal. Since in LES the large eddies are resolved explicitly on the spatial grid, only the effect of the removed small scales remains to be modeled. The more isotropic and (perhaps) universal nature of the small scales is a favorable point of departure for the modeling of the small-scale eddies. This modeling is referred to as subgrid-scale (SGS) modeling.

Before turning to the subject of SGS modeling, let us first consider the equations that need to be solved in an LES and estimate the associated computational effort. In an LES, a flow variable ϕ is decomposed in a grid-scale (GS) component $\overline{\phi}$ and a SGS component ϕ'. The GS component is defined by the moving average filter operation:

$$\overline{\phi}(x,t)=\int_V G(\mathbf{x}-\mathbf{x}')\phi(\mathbf{x}',t)d\mathbf{x}' \tag{8.7}$$

in which G is a spatial filter function that depends on the separation between **x** and **x'**, and V is the total volume of the computational domain. An often applied filter is the top-hat filter:

$$G(\mathbf{x}-\mathbf{x}')=\begin{array}{ll}\dfrac{1}{\Delta V} & \text{if}\quad |\mathbf{x}-\mathbf{x}'|\in \Delta V \quad \text{with} \quad \Delta V \propto l_\mathrm{f}^3 \\ & 0 \quad \text{elsewhere}\end{array} \tag{8.8}$$

Here l_f is a characteristic filter length. Application of the filter (Eq. 8.7) to the flow equations (Eqs. (8.1) and (8.2)) yields equations for the GS variables

$$\nabla\cdot\overline{\mathbf{v}}=0 \tag{8.9}$$

$$\frac{\partial\overline{\mathbf{v}}}{\partial t}+\nabla\cdot(\overline{\mathbf{v}}\,\overline{\mathbf{v}})=-\frac{1}{\rho}\nabla\overline{p}+\nu\nabla^2\overline{\mathbf{v}}-\nabla\cdot\boldsymbol{\tau} \tag{8.10}$$

where the continuity equations has been used in the momentum equation to reformulate the advective term in its conservative form. The stress $\boldsymbol{\tau}$ tensor represents the influence of the SGS motion on the GS motion. It reads

$$\boldsymbol{\tau} = \left(\overline{\overline{\mathbf{v}}\,\overline{\mathbf{v}}} - \overline{\mathbf{v}}\,\overline{\mathbf{v}}\right) + \left(\overline{\overline{\mathbf{v}}\,\mathbf{v}'} + \overline{\mathbf{v}'\,\overline{\mathbf{v}}}\right) + \overline{\mathbf{v}'\,\mathbf{v}'} \tag{8.11}$$

In a later stage, Eq. (8.11) will be discussed in more detail; for the moment it suffices to realize that $\boldsymbol{\tau}$ contains the unknown $\mathbf{v}'$ and requires modeling.

In LES, the macro-structure with length-scale l is still resolved on the spatial grid. The filter length l_f associated to the filter function $G(\mathbf{x} - \mathbf{x}')$ is now the characteristic length of the smallest resolved (GS) motions, as well as of the largest SGS motions. For LES to be realistic, the ratio l/l_f representing the range of length-scales of the GS motion should be $l/l_\mathrm{f}>1$ (smaller than one would imply that even the large-scale eddies of the macro structure are filtered out). The filter length l_f can be associated with the grid-spacing Δx where one usually assumes $l_\mathrm{f}=2\Delta x$ (Nyquist criterion). Larger values of l_f with unchanged Δx yield a more accurate numerical representation of the smallest resolved scales but simultaneously reduce the range of GS motions without any reduction of the computational effort. In LES one aims at keeping the range of GS motion as large as possible and therefore retains $l_\mathrm{f}=2\Delta x$. Thus, the linear grid-size N_L now becomes

$$N_L = \frac{L}{\Delta x} = \frac{L}{l}\frac{l}{l_\mathrm{f}}\frac{l_\mathrm{f}}{\Delta x} \approx 20\frac{l}{l_\mathrm{f}} \tag{8.12}$$

In contrast to Eq. (8.4), Eq. (8.12) is not related to Re_l anymore, but to the ratio l/l_f. As a result, LES is not restricted to low Re_l. The coefficient “20” in Eq. (8.12) appears due to the product L/l and $l_\mathrm{f}/\Delta x$. The parameter L/l depends on the type of flow. For shear driven flow it typically is 10. Apparently the ratio l/l_f is essential in LES. The choice of the ratio depends on the flow type under consideration. Roughly speaking, an LES with a value larger than 10 is considered to be well resolved.

Let us turn to the temporal resolution in LES. We have seen that the smallest resolved motions in LES are characterized by the filter length l_f. On similar grounds as before for the DNS, the time step Δt should thus be proportional to the time scale l_f/u. In practice, however, Δt in LES computations is determined by criteria for numerical stability that generally lead to small time steps such that the time-scale l_f/u is well-resolved.

8.3 Subgrid-scale modeling in LES

Due to the filtering operation, a stress tensor is introduced in the momentum equations which represents the effect of the SGS motion on the GS motion, see Eqs. (8.10) and (8.11). Here we repeat the expression for the stress tensor:

$$\boldsymbol{\tau} = \tau_{ij}\mathbf{e}_i\mathbf{e}_j = \left(\overline{\overline{\mathbf{v}}\,\overline{\mathbf{v}}} - \overline{\mathbf{v}}\,\overline{\mathbf{v}}\right) + \left(\overline{\overline{\mathbf{v}}\,\mathbf{v}'} + \overline{\mathbf{v}'\,\overline{\mathbf{v}}}\right) + \overline{\mathbf{v}'\,\mathbf{v}'} = \mathbf{L} + \mathbf{C} + \mathbf{R} \tag{8.13}$$

with τ_{ij} the components of $\boldsymbol{\tau}$, and **L**, **R** and **C** the Leonard stress, cross-terms, and SGS stress respectively. In most SGS modeling approaches, modeling applies to the tensor $\boldsymbol{\tau}$ as a whole. In that context $\boldsymbol{\tau}$ is simply referred to as the SGS stress. The tensor $\boldsymbol{\tau}$ is usually decomposed in an isotropic part and a non-isotropic part

$$\tau_{ij} = \frac{1}{3}\tau_{kk}\delta_{ij} + \left(\tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij}\right) \tag{8.14}$$

with δ_{ij} the Kronecker delta. The summation convention applies to the repeated index *k*. Usually, the isotropic part is combined with the resolved pressure. The non-isotropic part is denoted by $\boldsymbol{\tau}'$ and is thus given by

$$\boldsymbol{\tau}' = \tau'_{ij}\mathbf{e}_i\mathbf{e}_j = \left(\tau_{ij} - \frac{1}{3}\tau_{kk}\right)\mathbf{e}_i\mathbf{e}_j \tag{8.15}$$

Let us now consider modeling of $\boldsymbol{\tau}'$. Various approaches can be followed. Prognostic differential equations for all components τ'_{ij} can be derived form the momentum equations. In these equations, however, several terms appear which need to be parametrized, as they contain higher-order correlations of small scale fluctuations. Using this so-called second-order modeling would require solving 7 additional partial differential equations. Deardorff (1973) performed second-order LES for atmospheric turbulence. It yielded results superior to first-order approaches at increased computational cost. It is noteworthy that in later papers Deardorff reverted to much simpler gradient hypothesis approaches. A general simplification of second-order modeling is the reduction of partial differential equations into algebraic equations to parametrize the SGS stresses. It is believed that this approach retains the advantage of second-order modeling while it avoids an increase of computing time and memory usage. Applications of algebraic SGS models have been reported by Schem and Lipps (1976) and Schmidt and Schumann (1989) for studies of atmospheric turbulence. Of particular concern in second-order models is the attention that must be paid to the determination of the various coefficients involved (see also Schmidt and Schumann, 1989).

In view of the above remarks, one usually turns to first-order modeling in LES, where the SGS stresses are directly related to the GS velocity field in an algebraic way. To mimic the net energy transfer from large to small scales, the stress tensor $\boldsymbol{\tau}'$ is formulated in such a way that the SGS stresses reduce the kinetic energy of the GS velocity field. In this respect one introduces the so-called SGS eddy viscosity ν_e in analogy to the molecular viscosity to write $\boldsymbol{\tau}'$ as

$$\boldsymbol{\tau}' = -\nu_e\left(\nabla\overline{\mathbf{v}} + \left(\nabla\overline{\mathbf{v}}\right)^T\right) \tag{8.16}$$

Application of Eq. (8.16) requires a specification of ν_e. The most widely used model is the one proposed by Smagorinsky (1963) in which ν_e is related to the deformation of the resolved velocity field as

$$\nu_{\mathrm{e}} = l_{\mathrm{mix}}^2 \sqrt{\overline{S}^2} \quad \text{with} \quad \overline{S}^2 = \frac{1}{2}\left(\nabla\overline{\mathbf{v}} + (\nabla\overline{\mathbf{v}})^T\right) : \left(\nabla\overline{\mathbf{v}} + (\nabla\overline{\mathbf{v}})^T\right) \tag{8.17}$$

The length-scale l_{mix} represents the mixing length of the SGS motions and will be specified below.

The Smagorinsky model originates from the assumption of local equilibrium between production and dissipation in the equation governing the SGS kinetic energy E_{SGS} By definition:

$$E_{\mathrm{SGS}} = \int_{2\pi/l_{\mathrm{f}}}^{\infty} E(\kappa)\,\mathrm{d}\kappa \tag{8.18}$$

with $E(\kappa)$ the three-dimensional energy spectrum, κ the wavenumber and $2\pi/l_{\mathrm{f}}$ the smallest wavenumber corresponding to SGS motions. Local equilibrium between production and dissipation of E_{SGS}.is expressed by

$$-\left(\boldsymbol{\tau}' : \nabla\overline{\mathbf{v}}\right) = \varepsilon \tag{8.19}$$

which can be written with Eq. (8.16) as

$$\nu_{\mathrm{e}}\overline{S}^2 = \varepsilon \tag{8.20}$$

We now would like to relate the mixing length l_{mix} with the filter length l_{f}. For this reason the deformation rate $\overline{S}^2$ is formulated in terms of the energy spectrum $E(\kappa)$:

$$\overline{S}^2 = 2\int_0^{2\pi/l_{\mathrm{f}}} \kappa^2 E(\kappa)\,\mathrm{d}\kappa \tag{8.21}$$

This equation states that the resolved deformation rate $\overline{S}^2$ equals the dissipation spectrum $\kappa^2 E(\kappa)$ integrated over the resolved scales. For high Reynolds number turbulent flows and wavenumbers κ in the inertial subrange, the energy spectrum $E(\kappa)$ can be written as (e.g. Tennekes and Lumley, 1972)

$$E(\kappa) = \alpha_{\mathrm{K}}\varepsilon^{2/3}\kappa^{-5/3} \tag{8.22}$$

known as the $\kappa^{-5/3}$-law. The constant (attributed to Kolmogorov) α_{K} is approximately 1.6. Substitution of Eq. (8.22) in Eq. (8.18) and Eq. (8.21) respectively yields:

$$E_{\mathrm{SGS}} = \frac{2}{3}\alpha_{\mathrm{K}}\varepsilon^{2/3}\left(\frac{2\pi}{l_{\mathrm{f}}}\right)^{-2/3} \tag{8.23}$$

$$\overline{S}^2 = \frac{2}{3}\alpha_{\mathrm{K}}\varepsilon^{2/3}\left(\frac{2\pi}{l_{\mathrm{f}}}\right)^{4/3} \tag{8.24}$$

where we assumed Eq. (8.22) also for small wavenumbers (κ→0). This is not a valid assumption. However, $\kappa^2 E(\kappa)$ increases with increasing κ; and the contribution of the lower wavenumbers to the integral is relatively small. Substituting the expression for ν_{e} (Eq. 8.17), and the expression for ε that can be derived from Eq. (8.24) into Eq. (8.20) yields an expression for the ratio of mixing length and filter length:

$$\frac{l_{\mathrm{mix}}}{l_{\mathrm{f}}} = \frac{\left(\frac{3}{2}\alpha_{\mathrm{K}}\right)^{-3/4}}{2\pi} = 0.0825 \tag{8.25}$$

Let us now specify the mixing length. Similar to the filter length, the mixing length can be related to the grid spacing Δ (that for convenience is considered uniform and the same in all directions; a cubic grid). The ratio l_{mix} and Δ is denoted by the Smagorinsky constant c_{S}.:

$$c_{\mathrm{S}} = \frac{l_{\mathrm{mix}}}{\Delta} \tag{8.26}$$

If we assume $l_{\mathrm{f}}=2\Delta$, and use Eq. (8.25) we find $c_{\mathrm{S}} = 0.165$.

This value of the Smagorinsky constant was based on the equilibrium assumption, and on the existence of an inertial subrange ($\kappa^{-5/3}$ -law). Especially in wall bounded flows one finds that $c_{\mathrm{S}} = 0.165$ leads to overly damped simulations in which turbulence sometimes cannot be sustained. The remedy is to reduce c_{S} to values of typically 0.1. In principle this implies that the simulations are underresolved: $c_{\mathrm{S}} < 0.165$ can be interpreted as a filter length smaller than 2Δ.

The Smagorinsky model is a simple and robust model. It has two major drawbacks: it does not allow for backscatter (i.e. transfer of energy from the SGS to the GS), and it yields unrealistic results in the vicinity of walls. Since ν_{e} is a positive quantity, the effect of the SGS stresses in the momentum equations will always cause an energy transfer from GS to SGS, mimicking a cascade process. In real turbulence, there indeed is a net energy transfer from large to small scales. This net transfer, however, is the sum of forward scatter (GS to SGS) and backscatter (SGS to GS), where forward scatter dominates. In DNS it has been observed, however, that backscatter occurs in significant parts of a flow.

The Smagorinsky model has an isotropic ν_{e}. In particular near walls, ν_{e} should presumably be taken anisotropic which is not included in the standard Smagorinsky model. Also related to walls is the unphysical behavior of the SGS stress near walls. Precisely at walls, the SGS stress should vanish (no-slip condition), which is not guaranteed in the SGS model governed by Eqs. (8.16), (8.17), (8.26). A way to repair this is by means of wall-damping functions that reduce the mixing length towards the wall. An often-used damping function is due to Van Driest (1965) which relates l_{mix} to the (dimensionless) distance to the wall y^+:

$$l_{\mathrm{mix}} = c_{\mathrm{S}} \mathit{\Delta} \left(1 - e^{-y^{+}/A^{+}}\right) \tag{8.27}$$

The parameter

$$y^{+} = \frac{y u_{*}}{\nu} \tag{8.28}$$

is the wall unit or Reynolds number based on the distance to the wall y and the wall-shear velocity u_{*}. A^{+} is a constant equal to 26.

Dynamic subgrid-scale models (Germano et al, 1991) address the issues of walls and backscatter in the Smagorinsky model (see also the review article by Lesieur and Métais, 1996). In dynamic models the idea of an eddy viscosity related to the deformation rate, i.e. (Eqs. (8.17) and (8.26))

$$\nu_{\mathrm{e}} = c_{\mathrm{S}}^{2} \mathit{\Delta}^{2} \sqrt{\overline{S}^{2}} \tag{8.29}$$

is retained. The c_{S} in Eq. (8.29), however, is not a constant but is determined locally. It can become negative, thereby allowing for backscatter. In dynamic modeling, two filters are applied with different filter widths: *e.g.* $\mathit{\Delta}_{1}$ and $\mathit{\Delta}_{2}$ with $\mathit{\Delta}_{1} > \mathit{\Delta}_{2}$. Two eddy viscosities can now be determined:

$$\nu_{\mathrm{e1}} = c_{\mathrm{S}}^{2} \mathit{\Delta}_{1}^{2} \sqrt{\left(\overline{S}\right)_{1}^{2}} \quad \nu_{\mathrm{e2}} = c_{\mathrm{S}}^{2} \mathit{\Delta}_{2}^{2} \sqrt{\left(\overline{S}\right)_{2}^{2}} \tag{8.30}$$

with $\left(\overline{S}\right)_{1}$ and $\left(\overline{S}\right)_{2}$ the resolved deformation rates according to the two filters. The energy transfer (dissipation rate) at the two filter wavenumbers κ_{1} and κ_{2} is the same if the filter wavenumbers are chosen within the inertial subrange. Therefore (see Eq. 8.20)

$$\nu_{\mathrm{e1}} \left(\overline{S}\right)_{1}^{2} = \nu_{\mathrm{e2}} \left(\overline{S}\right)_{2}^{2} \tag{8.31}$$

The set of three equations (Eqs. 8.30 and 8.31) allows us to determine the unknowns $\nu_{\mathrm{e1}}, \nu_{\mathrm{e2}}$ and c_{S} . There are quite some practical issues in applying dynamic models. The most important is stability. Negative values of c_{S} (can) lead to negative viscosities and as a result unstable behavior of numerical schemes. Since the work of Germano et al. (1991) there have been extensive efforts in improving dynamic models.

8.4 Examples of single-phase LES by means of the lattice-Boltzmann method

The implementation of the concept of an eddy viscosity in the lattice-Boltzmann scheme is fairly straightforward. Instead of the molecular (constant) viscosity we substitute the sum of the molecular viscosity and the eddy viscosity in Eq. (4.28). The eddy viscosity varies in space and time and is determined via a SGS model.

If the viscosity is based on the local deformation rate, as is the case for the Smagorinsky model (see Eqs. (8.16) and (8.17)), an advantage of applying the LBM is that the deformation rate is readily available from the LB distribution functions (see Eq. (4.26)). This means that we do not need to calculate spatial derivatives of the velocity field (based on e.g. finite differences) to determine deformation rates.

What now follows are a few examples of LB/LES of flow systems typically encountered in (chemical) engineering. Such systems have complexly shaped (and sometimes moving) boundaries. In these situations, lattice-Boltzmann discretization of the flow equations can be a favorable approach in view of computational efficiency; in order to do realistic LES we need to have a sufficiently fine grid and numerical (parallel) efficiency is then of key importance. The examples are the flow in a mixing tank, and the flow in a swirl tube. In the latter case some aspects of SGS modeling will be highlighted.

Mixing tank simulation. Turbulently agitated tanks are used in various industries to perform mixing tasks in order to e.g. stimulate chemical reactions, bring species into contact, or disperse gases or solids into liquid. Predicting the single-phase flow in a mixing tank therefore is a relevant task. In order to prepare such a single-phase simulation for inclusion of chemical reactions, or solid particle dispersion, an LES approach is highly desirable. For instance: the dynamics of solid particles suspended in the turbulent flow is governed by their direct (hydrodynamic) surroundings that can be largely provided by an LES, not by the (time) averaged flow in the tank that would follow from a RANS-type of simulation. Here we report on simulations in a tank geometry as given in Figure 14. A revolving turbine with six vertical blades drives the flow. The vertically placed baffles at the perimeter of the tank enhance mixing as they largely prevent a solid-body rotation of the fluid. The Reynolds number of this flow is traditionally based on the impeller diameter D, and the angular velocity of the impeller N [in rev/s]: Re=D^2N/ν. Here, Re=$1\cdot10^5$. This flow has been simulated by means of lattice-Boltzmann discretization and a Smagorinsky subgrid-scale model with c_S=0.1. The uniform cubic mesh had a size of 240^3. The off-grid, no-slip boundary conditions at the tank wall, baffles, and revolving impeller were imposed by means of an immersed boundary technique that has been briefly described in Section 5.2. Further details of the simulations shown here can be found in e.g. Derksen and Van den Akker (1999), and Derksen (2003).

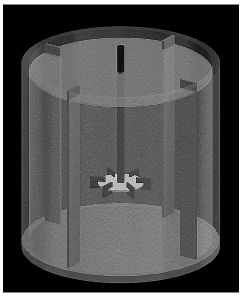

Figure 14. Typical mixing tank geometry: a baffled tank with a disk impeller (Rushton turbine) placed one-third of the tank height from the bottom.

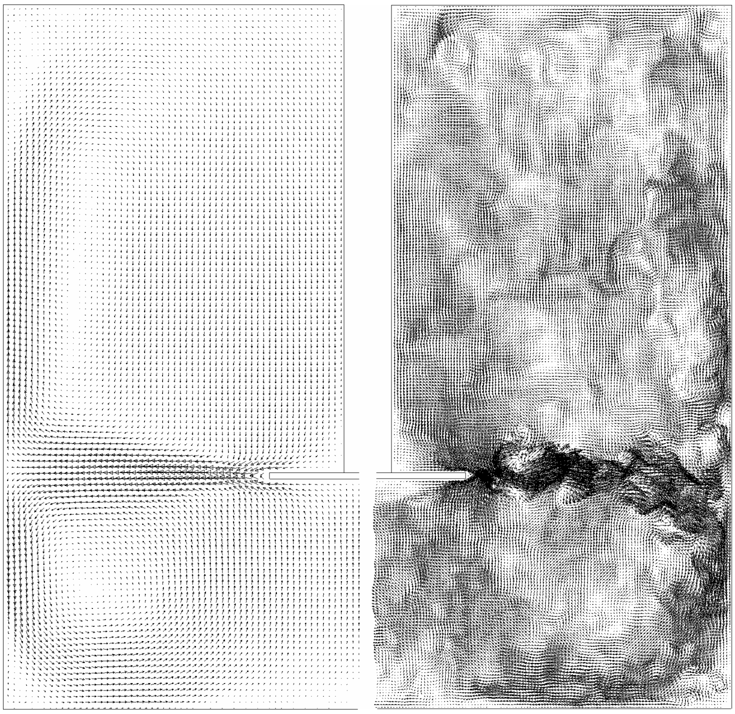

Figure 15. Flow field (in terms of velocity vectors) in a vertical plane through the center. Left: time-averaged flow; right: single realization.

In Figure 15, a snapshot and the average flow in a vertical cross section are compared. The average flow visualizes the action of the impeller: fluid is pumped in radial direction. Once it hits the outer wall, the fluid stream emerging from the impeller splits and drives two

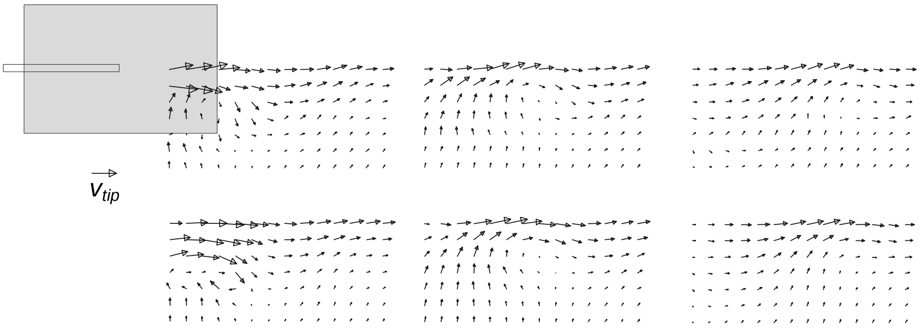

Figure 16. Comparison between experimental velocity data (top row, from Derksen et al 1999) and LES results (bottom row) at three angular positions in the wake of an impeller blade. From left to right the observation plane and the impeller blade make an angle of 10°, 31° and 49° respectively.

large recirculations, one above and one below the impeller. The slight upward inclination of the impeller stream is due to the impeller being placed closer to the bottom than to the top. This makes the lower recirculation stronger and pushes the impeller stream slightly upwards. A solid particle dispersed in the tank does not "see" the average flow. It is moved around by the instantaneous flow structures that are to a large extent resolved by the LES.

Validation of these simulations by means of experimental data is essential. Figure 16 shows a simulation-experiment comparison in a critical region of the flow: the wake of an impeller blade. The trailing vortex structure is well resolved by the simulations. Also the fluctuation levels (in terms of the turbulent kinetic energy) are well predicted by the LES (not shown here, see Derksen and Van den Akker, 1999).

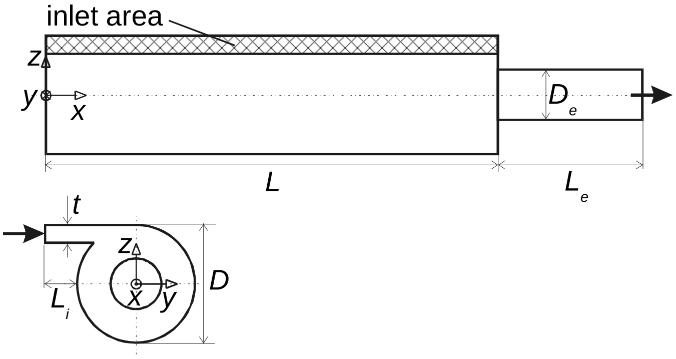

Figure 17. Flow geometry of the experimental setup due to Escudier et al' (1980).

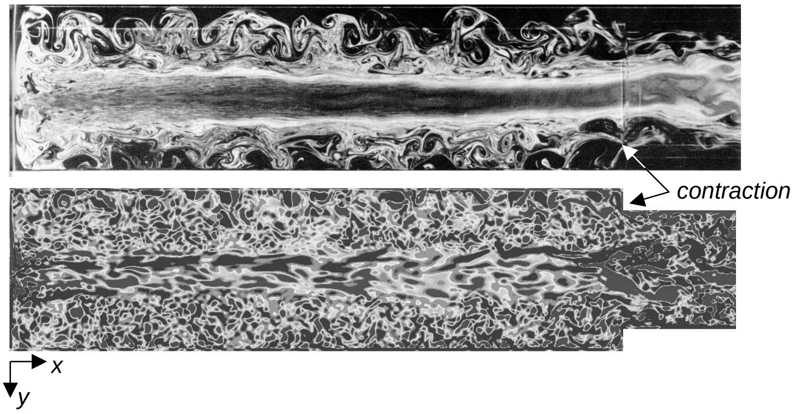

Figure 18. Top: (experimental) flow visualization due to Escudier et al' (1980) at Re=410, D_e/D=0.58. Bottom: LES snapshot at Re=2,100, D_e/D=0.73 in terms of the vorticity.

Swirl tube with tangential inlet. Turbulent swirling flows form a good testing ground for assessing simulation techniques and turbulence modeling approaches. This is because the turbulence is swirling flow is known to be highly anisotropic. Furthermore, swirl tends to

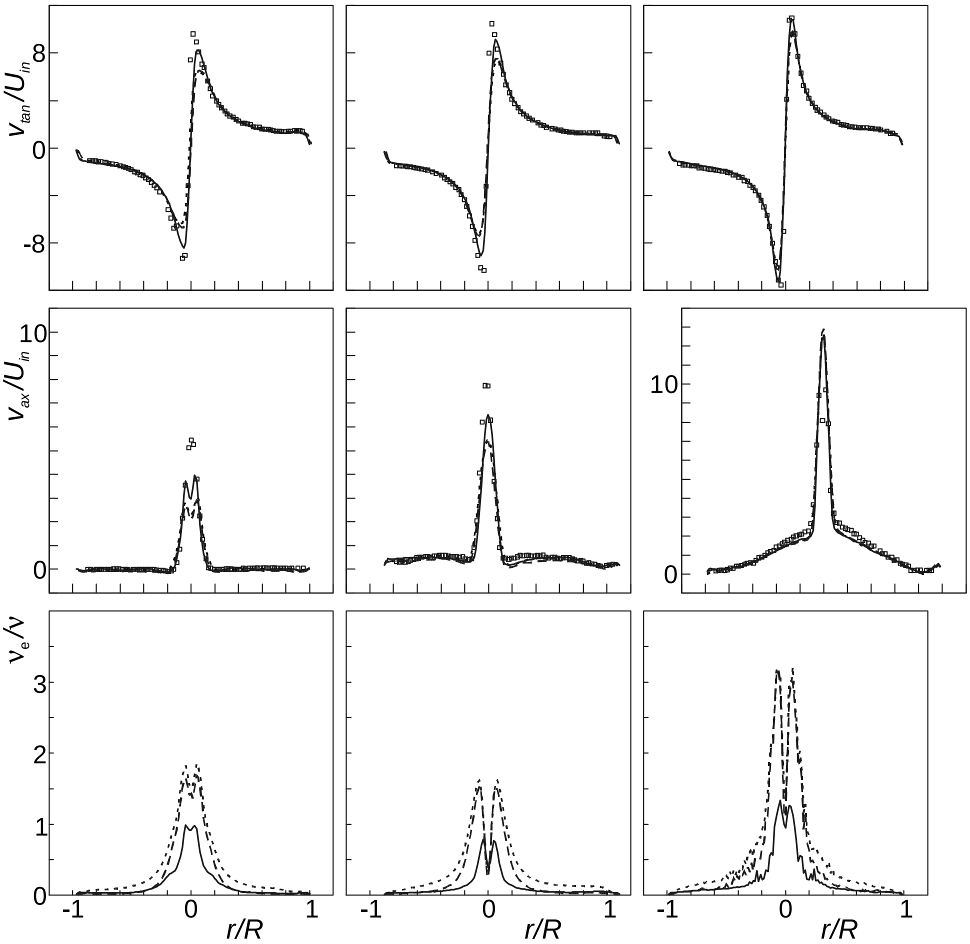

Figure 19. Average velocity and eddy-viscosity profiles for D_e/D=0.33. Top: tangential velocity; middle: axial velocity; bottom: eddy viscosity. From left to right: three axial positions x/D=0.15, 2.15, and 3.61. The symbols denote experimental data. The short-dashed line is the LES result with the Smagorinsky model, the long-dashed line with the model due to Voke, and the solid line with the mixed-scale model.

laminarize parts of turbulent flows. Swirling flows have practical significance in separation devices and combustion (flames stabilized by swirl). A classical experimental data set on turbulent swirling flow is due to Escudier and co-workers (see Escudier et al, 1980). The flow geometry is given in Figure 17. The complexity of the turbulence can be appreciated from Figure 18, which shows the flow visualized in an axial cross-section. A laminar-like vortex core can be distinguished from a turbulent outer region. Furthermore, vortex breakdown is observed (not shown here). Large-eddy simulations (Derksen, 2005) were able to represent these specific flow features very well (see Figure 18).

A quantitative comparison was made based on velocity profiles measured by Escudier et al (1980) by means of laser-Doppler anemometry (LDA). It was interesting to see the impact of the grid resolution, and the choice of the SGS model on the quality of the LES predictions. Next to the Smagorinsly model supplemented with Van Driest wall damping functions (Eq. (8.27)), we also applied a mixed-scale model due to Sagaut et al (2000), and a low-Reynolds number version of the Smagorinsky model due to Voke (1996). Some sample results are shown in Figure 19. They show that Sagaut et al's model is better capable of resolving the large gradients present in the flow. In relation to that, it can be seen that Sagaut et al's model has much lower eddy viscosities. Apparently the Smagorinsky model (and Voke's variant) produce relatively high eddy viscosities that smear out (dampen) steep flow gradients.

9 Point Particles in LES

In the previous section, two single-phase LES examples were discussed. In typical applications, however, mixing tanks and swirl separators are operated with multiple phases. Here we will discuss how to incorporate solid particles in LES flow fields. We will limit ourselves to a "point" particle approach. With this we mean that we do not resolve the flow around the particles with our lattice-Boltzmann discretization; the particles are smaller than the lattice spacing Δ (in Section 6, an approach is discussed where we do resolve the finite size of the particles; per particle such an approach requires much more computer resources than a point-particle approach and only small systems containing typically 10^3 particles can be simulated).

In a point-particle approach, the motion of the particles is determined by integrating the equation of motion of each individual particle (Lagrangian type simulations). The underlying fluid flow is based on an Eulerian (fixed-grid) method. Such Euler-Lagrange flow simulations are mostly limited to relatively dilute systems, i.e. systems with a low dispersed phase volume fraction. This is because of computational reasons: we can only track a limited number of particles that have relatively simple interactions; and because of reasons related to physical modeling: in dense suspensions the finite extent of the particles becomes relatively important and a point-particle approach cannot represent the physics very well. As an example: in fluidized beds (with typically 50% solids volume fractions) the fluid flow in between the solid particles is akin to the flow through a porous medium (i.e. flow through the channel-like structures in between particles) that is mimicked poorly by viewing the particles as points in space.

In the dilute limit amenable to an Euler-Lagrange approach, various subdivisions can be made, one of the most important being the choice of taking or not taking into account particle-particle collisions. In my view there is no general rule for taking into account collisions, especially in inhomogeneous flows with regions preferentially occupied by particles. In what follows we discuss a mixing tank containing a mixture of liquid and solid particles. The fluid flow is solved based on an LES approach and LB discretization (see Section 8.4). The geometry and fluid properties are the same as in the example discussed in that section.

In order to better appreciate the agitated solid-liquid system, we will now relate to dimensional numbers. The vessel volume was 10^{-2} m^3. This implies D=7.78·10^{-2} m. The continuous phase was water (with ν=1·10^{-6} m^2/s, and ρ_l=1·10^3 kg/m^3). A set of 6,705,623 spherical particles was released in the tank. The particles have a diameter of d_p=0.30 mm, and density ratio $\frac{\rho_\mathrm{p}}{\rho_\mathrm{l}}=2.5$ (with ρ_p the density of the particles), typical for glass beads in water. As a result, the solids volume and mass fractions amounted to Φ_V=0.95% and Φ_m=2.37% respectively. The impeller was set to revolve with N=16.5 rev/s (Re=1·10^5). The Stokes number of the particles was Stk=1.2 (with $\mathrm{Stk}=\frac{\rho_\mathrm{p}}{\rho_\mathrm{l}}\frac{d_\mathrm{p}^{\ 2}}{18\nu}6N$; the ratio of the Stokesian particle relaxation time, and the time of one impeller blade passage). The Stokes number is of the order of 1, and we may expect appreciable effects of particle inertia (i.e. the particles neither follow the flow (Stk<<1), nor move around in a ballistic manner (Stk>>1) and hardly feel the turbulent flow variations).

Without much discussion it can be anticipated that particle inertia and fluid inertia (added mass), gravity, and drag need to be part of the equations of motion of the solid particles. Since a stirred tank flow is very inhomogeneous, it is difficult to estimate *a priori* if more exotic forces like lift and history forces play an important role. For instance, an estimate of the ratio between lift (Magnus and/or Saffman force) and drag forces is $0.2\sqrt{\frac{d_\mathrm{p}^{\ 2}|\hat{\boldsymbol{\omega}}|}{\nu}}$, with $|\hat{\boldsymbol{\omega}}|$ the vorticity in the liquid phase or the angular (slip) velocity of the particle. In the impeller region of single-phase stirred tank flow, vorticity easily exceeds 10N (Derksen and Van den Akker, 1999). With N=16.5 rev/s, and d_p=0.3 mm the ratio amounts to 0.8, indicating the potential relevance of lift forces.

Each particle dispersed in the stirred tank has six degrees of freedom associated to it: three linear coordinates, and three angles. Since we consider the particles to be spherical, the particle's orientation has no physical consequence. As will be demonstrated below, the angular velocity has physical significance. For the linear motion, the following set of equations will to be solved:

$$\frac{\mathrm{d}\mathbf{x}_\mathrm{p}}{\mathrm{d}t}=\mathbf{v}_\mathrm{p} \tag{9.1}$$

$$\frac{\pi}{6}d_\mathrm{p}^{\ 3}\left(\rho_\mathrm{p}+\frac{1}{2}\rho_\mathrm{l}\right)\frac{\mathrm{d}\mathbf{v}_\mathrm{p}}{\mathrm{d}t}=\frac{\pi}{8}d_\mathrm{p}^{\ 2}\rho_\mathrm{l}C_\mathrm{D}\left|\mathbf{v}-\mathbf{v}_\mathrm{p}\right|\left(\mathbf{v}-\mathbf{v}_\mathrm{p}\right)+\mathbf{F}_\mathrm{Saffman}+\mathbf{F}_\mathrm{Magnus}+\mathbf{F}_\mathrm{stress}+\frac{\pi}{6}d_\mathrm{p}^{\ 3}\left(\rho_\mathrm{p}-\rho_\mathrm{l}\right)\mathbf{g} \tag{9.2}$$

with $\mathbf{x}_\mathrm{p}$ the center position of the particle, $\mathbf{v}_\mathrm{p}$ and $\mathbf{v}$ the velocity of the particle, and the velocity of the liquid at $\mathbf{x}_\mathrm{p}$ respectively, and C_D the drag coefficient. The latter depends on the particle Reynolds number $\mathrm{Re}_\mathrm{p}=\frac{\left|\mathbf{v}-\mathbf{v}_\mathrm{p}\right|d_\mathrm{p}}{\nu}$ according to Eq. (A.1), given in the *Appendix* (Sommerfeld, 2001). The solids volume fractions are considered to be sufficiently low not to include a dependency of C_D on Φ_V. Added mass is accounted for by the additional particle inertia $\frac{\pi}{12}d_{\mathrm{p}}{}^{3}\rho_{\mathrm{l}}$ (Maxey and Riley, 1983). The influence of the Saffman force ($\mathbf{F}_{\mathrm{Saffman}}$), Magnus force ($\mathbf{F}_{\mathrm{Magnus}}$), and the force due to stress gradients ($\mathbf{F}_{\mathrm{stress}}$) will be discussed below where simulations with and without these forces will be compared.

The Basset history force (Odar and Hamilton, 1964) may have some impact in the impeller region, with its strong velocity fluctuations at frequencies of the order of 6*N*. The ratio between the Basset history force and Stokes drag in a time-varying flow field with frequency *f* is of the order of $0.1\sqrt{\frac{d_{\mathrm{p}}{}^{2}f}{\nu}}$. If we take *f*=6*N*, the ratio is 0.3. It will be demonstrated, however, that in the impeller region, Re_{p} is of the order 10^2. As a result, the drag force is one order of magnitude higher than estimated from Stokes drag, and the Basset force becomes small compared to the drag force. For this reason, and for computational reasons (inclusion of the force would add appreciably to the computational effort), the Basset force has been neglected.

The non-Stokes expressions we use for the Saffman and Magnus force respectively are (Mei, 1992; Oesterlé and Bui Dinh, 1998)

$$\mathbf{F}_{\mathrm{Saffman}} = \frac{\pi}{4}d_{\mathrm{p}}{}^{3}\frac{\rho_{\mathrm{l}}}{2}C_{\mathrm{S}}\left(\left(\mathbf{v}-\mathbf{v}_{\mathrm{p}}\right)\times\boldsymbol{\omega}\right) \tag{9.3}$$

$$\mathbf{F}_{\mathrm{Magnus}} = \frac{\pi}{4}d_{\mathrm{p}}{}^{2}\frac{\rho_{\mathrm{l}}}{2}C_{\mathrm{M}}\left|\mathbf{v}-\mathbf{v}_{\mathrm{p}}\right|\frac{\left(\boldsymbol{\omega}-2\boldsymbol{\omega}_{\mathrm{p}}\right)\times\left(\mathbf{v}-\mathbf{v}_{\mathrm{p}}\right)}{\left|\boldsymbol{\omega}-2\boldsymbol{\omega}_{\mathrm{p}}\right|} \tag{9.4}$$

with $\boldsymbol{\omega}$ the vorticity of the liquid, and $\boldsymbol{\omega}_{\mathrm{p}}$ the angular velocity of the particle. The lift coefficients C_{S} and C_{M} depend on Re_{p}, and on the rotational Reynolds numbers $\mathrm{Re}_{\mathrm{S}} = \frac{|\boldsymbol{\omega}|d_{\mathrm{p}}{}^{2}}{\nu}$, and $\mathrm{Re}_{\mathrm{R}} = \frac{\left|\frac{1}{2}\boldsymbol{\omega}-\boldsymbol{\omega}_{\mathrm{p}}\right|d_{\mathrm{p}}{}^{2}}{\nu}$ according Eqs. (A.2) and (A.3). The force due to stress gradients has a pressure and a viscous stress part:

$$\mathbf{F}_{\mathrm{stress}} = \frac{\pi}{6}d_{\mathrm{p}}{}^{3}\left(-\nabla p+\rho_{\mathrm{l}}\nu\nabla^{2}\mathbf{v}\right) \tag{9.5}$$

In order to determine the Magnus force (Eq. (9.4)), the angular velocity of the particles needs to be solved. This is done by solving the following dynamic equation:

$$\frac{\mathrm{d}\boldsymbol{\omega}_{\mathrm{p}}}{\mathrm{d}t} = \frac{60}{d_{\mathrm{p}}{}^{2}}\frac{\rho_{\mathrm{l}}}{\rho_{\mathrm{p}}}\nu\left(\frac{1}{2}\boldsymbol{\omega}-\boldsymbol{\omega}_{\mathrm{p}}\right) \tag{9.6}$$

which is valid for $\mathrm{Re}_{\mathrm{R}} \le 30$ (Dennis et al., 1980). The particles' angular velocity may also be relevant from a practical point of view. Mass transfer between solid particles and continuous phase liquid depends on the motion of the solid surface relative to the liquid. Apart from linear velocities, particle rotation might play a role in mass exchange.

The fluid's velocity $\mathbf{v}$, vorticity $\boldsymbol{\omega}$, pressure p and viscous stress $\rho_l \nu \nabla \mathbf{v}$ contained in the above equations all consist of a resolved and a subgrid-scale (SGS) part. For reasons of simplicity, the SGS parts have been discarded, except when the drag force is involved. For determining the drag force, the local fluid velocity is considered to be the sum of the resolved velocity and a Gaussian random process with standard deviation $u_{\mathrm{sgs}} = \sqrt{\frac{2}{3} k_{\mathrm{sgs}}}$ representing the SGS motion. The SGS kinetic energy k_{sgs} was estimated based on isotropic, local-equilibrium mixing-length reasoning according to

$$k_{\mathrm{sgs}} = C_{\mathrm{k}} c_{\mathrm{s}}^2 \Delta^2 \bar{S}^2 \tag{9.7}$$

with C_{k} a constant amounting to 5 (Mason and Callen, 1986). To have temporal coherency in the SGS motion, a new random velocity was picked after the elapse of a SGS eddy lifetime

$$t_{\mathrm{sgs}} = C_{\mathrm{L}} \frac{k_{\mathrm{sgs}}}{\varepsilon} \tag{9.8}$$

with the constant C_{L}=0.15 (Weber et al. 1984), and ε the energy dissipation rate.

The resolved part of the liquid velocity was determined by linearly interpolating the velocities on the lattice-Boltzmann grid to the particle position (although a higher order interpolation scheme may be needed for accurate evaluation of single- and, especially, two-particle Lagrangian statistics; see, for example Kontomaris et al. 1992). The vorticity, and the pressure and viscous stress gradients felt by the solid particles were taken uniform over a grid cell. The stresses are directly contained in the solution vector of the lattice-Boltzmann scheme. Their gradients, as well as the velocity gradients contained in the vorticity were determined from central finite differencing. Note that the determination of the force due to stress-gradients differs from how it is usually done, i.e. by means of the material time derivative of the fluid velocity (see e.g. Crowe et al. 1998). Since in a lattice-Boltzmann scheme the stresses are readily available, it is not expensive to directly determine the stress gradients.

Two types of collisions need to be distinguished: particle-wall, and particle-particle collisions. Collisions of all types were considered to be fully elastic and frictionless (the latter implies that in a particle-wall collision the wall parallel components of the velocity of the particle *surface* are unchanged after a collision; in a particle-particle collision the rotation of the particles does not play a role in the collisional process). For the particle-wall collisions with the wall being part of the impeller, only a collision with one of the (six) impeller blades adds momentum to a particle since only the impeller blades have a velocity component in their wall-normal direction.

The method for detecting and handling particle-particle collisions was similar to the one proposed by Chen et al. (1998). In their method, they make use of a collision detection algorithm that anticipates collisions in the upcoming time step. Subsequently, the path of two particles that are bound to collide is integrated in a three-step-process: the pre-collision step, the collision step (in which the particles exchange momentum), and the post-collision step. In order to limit the computational effort spent in handling the particle-particle collisions (which in principle is an M^2 process, with M the number of particles) we have grouped the particles in each other's vicinity in a so-called link-list (Chen et al., 1998). The extent of the vicinity of a particle in which potential collision partners are sought is the lattice cell in which the particle under consideration resides, and the 26 neighboring cells. The distance traveled by a particle during one time step was at most 0.2Δ. This reduces the number of possible collisions partners to a few for a specific particle during a specific time-step. The collision algorithm assumes that one particle can only collide once during one time step. The reason is purely practical: taking into account multiple collisions in one time step would lengthen the computations to an unfeasible extent (e.g. allowing for the possibility to have two collisions per particle per time step would make an M^3 process). The assumption either limits the time step, or the particle volume fraction. In any case, in the simulations there is a finite chance that the collision detection algorithm misses a collision. This is reflected in the situation that at the next time step, two approaching particles have a mutual distance less than d_p. If this occurs, a so-called missed collision procedure is executed: directly at the start of the time-step, the particles involved are given their post-collision velocities (making that they now are moving apart). During the time step, the particles are displaced as a pair according to their average veloc-

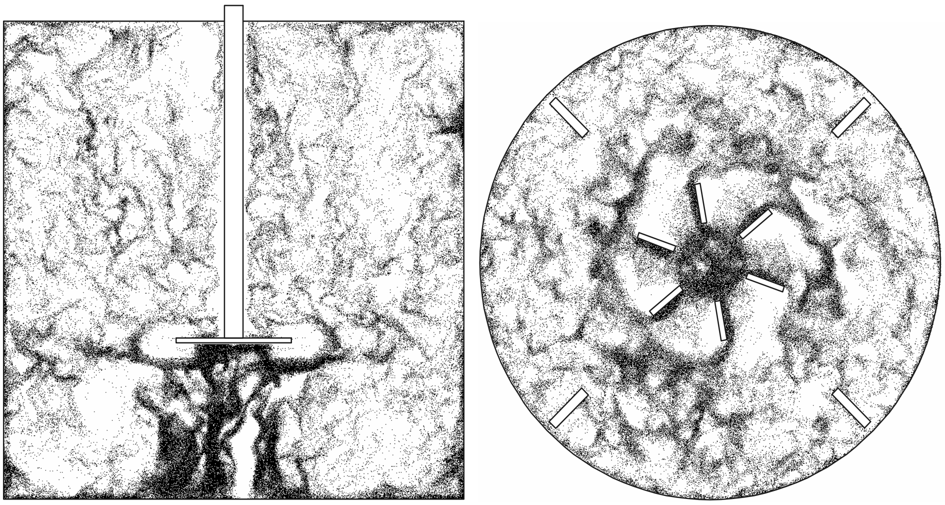

Figure 20. Instantaneous realization of the particle distribution in the tank. Left: vertical cross section through the center of the tank midway between two baffles; bottom: horizontal cross section at z/T=0.308 (i.e. just below the impeller disk). The impeller rotates in the counter-clockwise direction. In both graphs, the particles in a slice with thickness 0.0083T have been displayed.

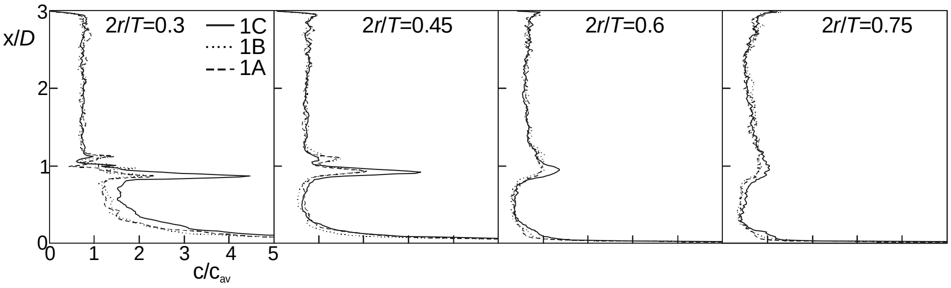

Figure 21. Phase-averaged particle concentration profiles in the plane midway between two baffles as a function of the vertical position in the tank at four different radial positions. Comparison between case #1A (only drag and gravity), #1B (plus lift forces), and #1C (plus particle-particle collisions).

ity, and they move apart with their relative velocity until they have a mutual separation of at least d_p.

The particle-particle collision algorithm has been tested by (numerically) releasing a set of particles with random initial velocity (according to a uniform distribution) in a periodic vacuum box. The velocity distribution should relax to a Maxwellian distribution, which it did. In the same setup, the algorithm described above to repair missed collisions was tested. Ignoring missed collisions led to one order of magnitude more overlapping particles at any moment in time compared to a situation in which the missed collision procedure was applied.

In the simulations, solids volume fractions are such that it is expected that two-way coupling effects are relevant (Elgobashi, 1994). Two-way coupling was achieved by feeding the force that the fluid exerts on the particle back to the fluid. Since the center position of a particle in general does not coincide with a grid point, the back-coupling force needs to be distributed over the (lattice-Boltzmann) grid nodes in the vicinity of the particle (particle-source-in-cell (PSIC) method, see Crow et al, 1996). For this extrapolation we used the same coefficients as were used for the linear interpolation of the velocity at the grid nodes to the particle location.

Some impressions of these Euler-lagrange two-phase flow simulations are given in Figures 20 and 21. In Figure 20 we see the strongly inhomogeneous distribution of particles throughout the tank. In the vertical cross-section, a highly concentrated region underneath the impeller can be observed. Here gravity and an upwardly directed flow somehow balance the particle motion. This gives rise to long residence times there. The streaky patterns are due to particles collecting at the edges of eddies. This is a typical phenomenon for particles with Stokes numbers of the order of one. Smaller particles would show a much more homogeneous distribution. Particle inertia is also apparent from the horizontal cross-section: particles collect in front of the impeller blades while the wakes of the blades are almost void of particles. In Figure 21 particle concentration profiles are presented obtained with different modeling assumptions. The most striking feature is the impact of taking into account particle-particle collisions has on the concentration profiles. A more detailed analysis reveals that it is the volume exclusion effect brought about by the collision algorithm that makes the difference. If particle-particle collisions are not taken into account, unrealistically high particle concentrations closely above the bottom of the tank are observed. Volume exclusion reduces these concentrations strongly. The profiles obtained with particle-particle collisions fairly well agree with experimental data (Derksen, 2003); the strong peak at the impeller level has also been observed in experiments.

10 Passive Scalar Transport

In order to describe passive scalar transport in a laminar or turbulent flow, the convection-diffusion-equation needs to be solved

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = -\nabla \cdot \mathbf{J} \tag{10.1}$$

with ϕ the scalar concentration, and **J** the diffusive mass flux. In many cases Fick's law applies:

$$\mathbf{J} = -\Gamma \nabla \phi \tag{10.2}$$

with Γ the scalar's diffusivity. The scalar transport has its own micro length and time-scales. Of prime importance in this respect is the Batchelor length-scale:

$$\eta_{\mathrm{B}} = \left(\Gamma^2 \frac{\nu}{\varepsilon} \right)^{1/4} = \eta \mathrm{Sc}^{-1/2} \quad \text{with} \quad \mathrm{Sc} = \frac{\nu}{\Gamma} \quad \text{the Schmidt number} \tag{10.3}$$

The Batchelor scale can be interpreted as the diffusion distance during one Kolmogorov time scale. Now an essential difference between gases and liquids can be appreciated. In gases the Schmidt number is of the order 1 (momentum and species diffuse at approximately the same pace); in liquids the Schmidt number is $O(10^3)$ (species diffuse much slower than momentum). In a direct numerical simulation of turbulent flow and associated scalar transport, the grid needs to be fine enough to resolve all scales. If the medium is a gas, the grid that was used to resolve the gas motion is sufficiently fine to also resolve the concentration field. If the medium is a liquid, a DNS resolving the all scales of the concentration field would require a grid that (in linear terms) is of the order of $\sqrt{1000} \approx 30$ times finer than the grid required for flow simulations.

Suppose we perform an LES, and would like to represent the scalar concentration fields on the same grid as the velocity and pressure field. This implies filtering of Eq. (10.1):

$$\frac{\partial \overline{\phi}}{\partial t} + \nabla \cdot \left(\overline{\mathbf{v}}\, \overline{\phi} \right) = \Gamma \nabla^2 \overline{\phi} - \nabla \cdot \boldsymbol{\sigma} \tag{10.4}$$

with $\boldsymbol{\sigma}$ the equivalent of $\boldsymbol{\tau}$ in Eq. (8.10). Usually $\boldsymbol{\sigma}$ is closed in a manner similar to the closure of $\boldsymbol{\tau}$: we assume that the subgrid-scales merely act in a diffusive manner on the concentration field. The eddy diffusion coefficient is then taken proportional to the eddy-viscosity:

$$\Gamma_{\mathrm{e}} = \frac{\nu_{\mathrm{e}}}{\mathrm{Sc}_{\mathrm{e}}} \tag{10.5}$$

with Sc_{e} the turbulent (eddy) Schmidt number. For Sc_{e} a value in the range of 0.6 to 1.0 is usually taken. The rationale behind $\mathrm{Sc}_{\mathrm{e}} < 1$ (at least in liquid systems) is that the unresolved part of the scalar spectrum is larger (it runs up to $\kappa \approx {}^{2\pi}\!/\!_{\eta_{\mathrm{B}}}$) than the unresolved part of the turbulent kinetic energy spectrum.

Eggels and Somers (1995) have performed scalar transport calculations on free convective cavity flow with the lattice-Boltzmann discretization scheme (i.e. they solved the fluid flow equations, and the convection diffusion equation with the lattice-Boltzmann method). This approach, however, is more memory intensive than using a finite volume formulation for the convection-diffusion equation. In a finite volume discretization we only need to store two or three (depending on the time integrator) double precision concentration fields,

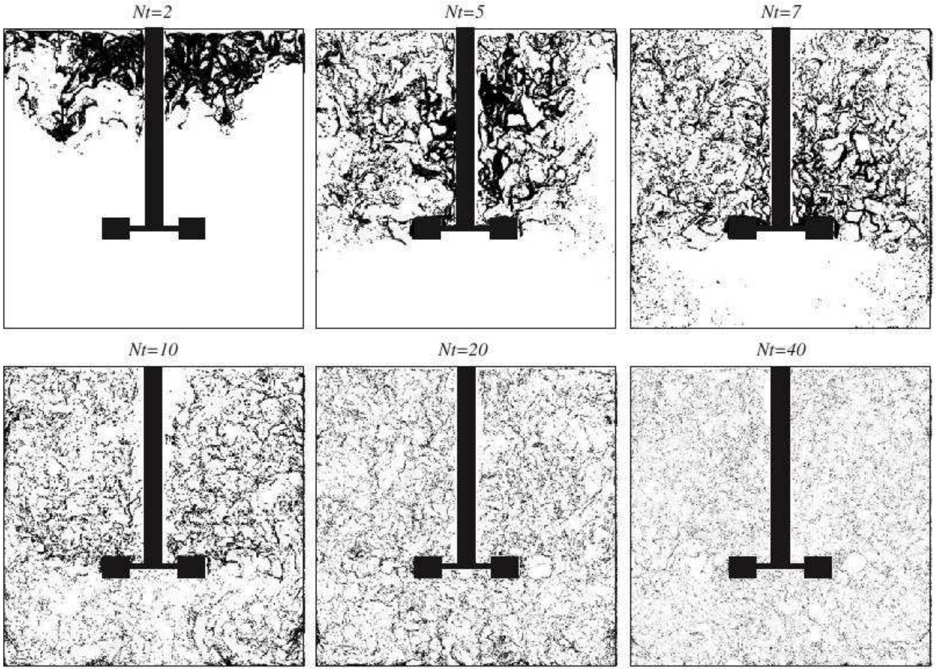

Figure 22. Snapshot of the particle distribution in a vertical plane midway between two baffles.

whereas in the lattice-Boltzmann discretization typically 18 single-precision variables need to be stored. In a time-explicit approximation of the discretized convection-diffusion equation we have (as in the lattice-Boltzmann scheme) fully local operations: The communication between the nodes of the grid defining the concentration field does not go beyond the stencil that is used for discretizing the convection and diffusion term in Eq. (10.4). An additional advantage of using finite volume discretization for the convection diffusion equation is that we can make use of the various methods to suppress numerical diffusion and still retain stability, such as TVD (total variation diminishing) schemes introduced by Harten (1983).

As an example of LES including scalar transport and solid particle dynamics we briefly discuss here work presented earlier by Hartmann et al (2006) on a dissolution process in a mixing tank. In this example we combine the solid-liquid LES based on the point particle approach (see Section 9) with a finite-volume scalar transport solver for keeping track of the scalar concentration as a result of dissolution. The coupling between the scalar field and the solids is established via a mass transfer coefficient k, which is a function of the particle size, the relative velocity of particle and fluid $\mathrm{v}_{\mathrm{slip}}$, and the material properties (viscosity and diffusivity) of the fluid. In dimensionless form this implies that the Sherwood number $\mathrm{Sh} = kd_{\mathrm{p}} / \Gamma$ is a function of the particle Reynolds number and the Schmidt number. We use the correlation due to Ranz and Marshall (1952):

$$\mathrm{Sh} = 2.0 + \mathrm{Re}_{\mathrm{p}}^{1/2}\,\mathrm{Sc}^{1/3} \qquad (10.6)$$

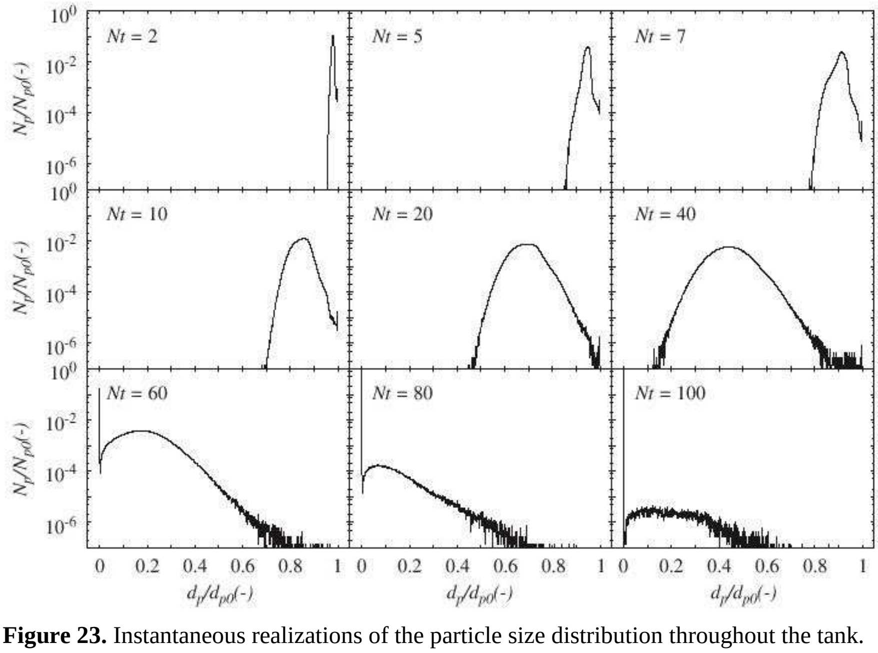

Figure 23. Instantaneous realizations of the particle size distribution throughout the tank.

The mass transfer from solid to liquid is then modeled as

$$\dot{m} = k\pi d_{\mathrm{p}}^{2}\left(c_{\mathrm{sat}} - c\right) \tag{10.7}$$

Since this relation is linear in the solute concentration, micro-mixing effects are considered not important (there is, however, a velocity-concentration correlation since k is a function of Re that in principle requires SGS closure; here this correlation is neglected).

The tank that Hartmann et al (2006) used was the same as the one discussed in the previous sections. Particles of uniform size were released in the top part of the tank in a fully developed flow field. They spread through the tank and reduce in size (see Figure 22). As they reduce in size they go through a spectrum of Stokes numbers, and the particle field gradually loses its streaky structure; particles get more homogeneously dispersed over the entire tank (Figure 22). The simulations allow for an estimate of the dissolution time. The evolution of the particle size distribution (Figure 23) reflects the diversity in the history of the dissolution process per particle, and thereby the inhomogeneous flow conditions in the tank.

11 Filtered Density Functions for Reactive Flows

In the final section of this chapter, we consider reactive scalar transport. In that case, the species transport equation (the convection diffusion Eq. (10.1)) is adapted in two ways. In the first place, instead of a single scalar ϕ we now write the vector $\boldsymbol{\varphi}$, reflecting that we have a number of species involved in the reactions that can be organized in a vector. In the second place, concentrations can now change as a result of reactions: we need to add a reaction term $\boldsymbol{\omega}$ that (in the general case) depends on all species involved (i.e. the vector $\boldsymbol{\varphi}$):

$$\frac{\partial \boldsymbol{\varphi}}{\partial t} + \mathbf{v} \cdot \nabla \boldsymbol{\varphi} = -\nabla \cdot \mathbf{J} + \boldsymbol{\omega}(\boldsymbol{\varphi}) \tag{11.1}$$

In the case of second order reactions, $\boldsymbol{\omega}$ depends on products of the elements of the vector $\boldsymbol{\varphi}$. In an LES context, Eq. (11.1) could be filtered just as Eq. (10.1). The result is (again we have assumed Fickian diffusion for all species involved)

$$\frac{\partial \overline{\boldsymbol{\varphi}}}{\partial t} + \nabla \cdot \left(\overline{\boldsymbol{\varphi}}\,\overline{\mathbf{v}}\right) = \Gamma \nabla^{2} \overline{\boldsymbol{\varphi}} + \overline{\boldsymbol{\omega}}(\boldsymbol{\varphi}) - \nabla \cdot \boldsymbol{\sigma} \tag{11.2}$$

where $\boldsymbol{\sigma}$ now is a tensor. The filtered reaction term $\overline{\boldsymbol{\omega}}(\boldsymbol{\varphi})$ needs specific care. It would be too gross a simplification to write $\overline{\boldsymbol{\omega}}(\boldsymbol{\varphi}) = \boldsymbol{\omega}(\overline{\boldsymbol{\varphi}})$. Remember that we are performing an LES. It only makes sense to perform an LES if the grid is courser than the Kolmogorov scale. This implies that also the Batchelor scale (the smallest scale of scalar transport) is not resolved. As a result, the concentration field shows details finer than the grid spacing. If we would state $\overline{\boldsymbol{\omega}}(\boldsymbol{\varphi}) = \boldsymbol{\omega}(\overline{\boldsymbol{\varphi}})$, this would imply the assumption that at the grid level species concentrations are uniform, and this is not the case.

The filtered reaction term $\overline{\boldsymbol{\omega}}(\boldsymbol{\varphi})$ is known once the filtered version of the probability density function $P_{\mathrm{L}}(\boldsymbol{\varphi};\mathbf{x},t)$ (the fdf, filtered density function) of the vector $\boldsymbol{\varphi}$ at the nodes of the computational domain is known. For example, for a second order reaction *A*+*B*→*C* with reaction rate *k*

$$\overline{\boldsymbol{\omega}}(\boldsymbol{\varphi}) = \int k\psi_{\mathrm{A}}\psi_{\mathrm{B}}P_{\mathrm{L}}(\psi_{\mathrm{A}},\psi_{\mathrm{B}})d\psi_{\mathrm{A}}d\psi_{\mathrm{B}} \qquad (11.3)$$

where the integration is over the entire (in this case two-dimensional) composition space.

This would imply that, instead of solving transport equations for the species concentrations $\boldsymbol{\varphi}$, we would need to solve transport equations for $P_{\mathrm{L}}(\boldsymbol{\varphi};\mathbf{x},t)$. This may seem a quite impossible task since the dimensionality of the system of equations to be solved increases rapidly. Apart from the three spatial dimensions the composition space adds to the dimensionality. With each species the dimension of the problem increases with 1. A way around this is to solve the fdf-transport equations by means of a Monte-Carlo (MC) method, i.e. to release computational particles in the flow domain and track their position in spatial and compositional space, the major advantage being that adding a reactant increases the computational load approximately linearly.

The idea is to release MC particles randomly in the computational domain. Each particle represent the scalar composition **φ** at its current position **x**(t). The MC particle position and composition are evolved according to the following stochastic differential equations

$$\mathrm{d}\mathbf{x} = \mathbf{D}(\mathbf{x}(t),t)\mathrm{d}t + E(\mathbf{x}(t),t)\mathrm{d}\mathbf{W}(t) \quad \text{and} \quad \mathrm{d}\boldsymbol{\varphi} = \mathbf{B}(\boldsymbol{\varphi}(t),t)\mathrm{d}t \qquad (11.4)$$

where **D** and *E* are the drift (convection) and diffusion coefficients of the particles in the physical domain. The random process $\mathrm{d}\mathbf{W}$ is a Wiener process; $\mathrm{d}\mathbf{W} = \mathrm{d}W_i = \sqrt{\mathrm{d}t}\,\zeta_i$ with ζ_i a random variable with Gaussian pdf. The drift **B** in the scalar domain is due to micro-mixing and chemical reactions. The various processes in Eq. (11.4) can be related to physical quantities:

$$E = \sqrt{2(\Gamma + \Gamma_{\mathrm{e}})}\,, \quad \mathbf{D} = \overline{\mathbf{v}} + \nabla(\Gamma + \Gamma_{\mathrm{e}}) \quad \text{and} \quad \mathbf{B} = -\Omega_{\mathrm{m}}(\boldsymbol{\varphi} - \overline{\boldsymbol{\varphi}}) + \boldsymbol{\omega}(\boldsymbol{\varphi})$$

where as a micro-mixing model we have substituted the interaction-by-exchange-with-the-mean (IEM) model. The SGS mixing frequency Ω_{m} can be related to the total (molecular plus eddy) diffusivity $\Gamma + \Gamma_{\mathrm{e}}$:

$$\Omega_{\mathrm{m}} = \frac{C_{\Omega}(\Gamma + \Gamma_{\mathrm{e}})}{\Delta^2} \qquad (11.5)$$

with Δ the filter width, and C_{Ω} a constant equal to 3 (Colucci et al 1998).

As an example of the application of the LES/FDF approach to a semi-practical system, some results earlier presented by Van Vliet et al (2005) are shown here. Van Vliet et al performed an LES in a straight tube with a deeply protruding feed pipe. The Reynolds number of the main flow was 4,000. Upstream of the feed pipe, the main flow contained species *B* and *C* in the same amount. Component *A* was fed through the feed pipe. *A* could react either

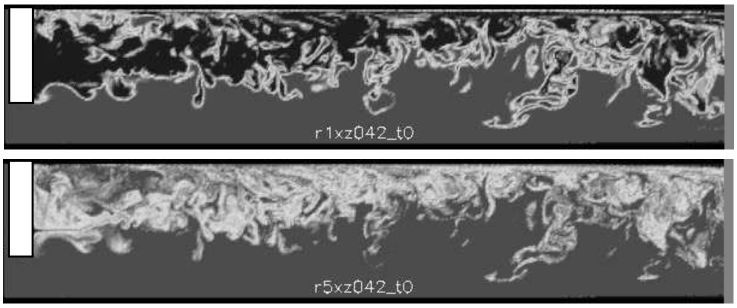

Figure 24. Instantaneous realizations of c_B in the tube reactor. Top: at Da=2.5·10^3 (poor mixing); bottom: Da=2.5·10^{-5} (intense mixing).

with *B* or *C* according to a second-order reaction to form product *P* or *Q* respectively. The reaction rate to form *P*, however, was 10^3 higher than the one to form *Q*. If the chemical kinetics would control the system, the amount of *Q* formed (Φ_Q) would be a thousand times smaller than the amount of *P* formed (Φ_P). Non-ideal mixing, however, will generally increase the ratio Φ_Q/Φ_P: The reaction between *A* and *B* locally depletes the flow of *B*. If mixing cannot bring *A* into contact with fresh *B* quickly enough, the slow reaction will get a chance.

The strength of mixing has been quantified in terms of a Damkohler number, which we define as the ratio of the integral hydrodynamic time-scale *D*/*U* (with *D* and *U* the tube diameter and bulk velocity respectively), and the chemical time scale $1/\sqrt{k_2 c_{A0} c_{C0}}$ with k_2 the rate constant of the slow reaction, and c_{A0} and c_{C0} the inlet concentrations:

$$\mathrm{Da} = \frac{D\sqrt{k_2 c_{A0} c_{C0}}}{U} \qquad (11.6)$$

Some results of the simulations are shown in Figures 24 and 25. In Figure 24 we see that

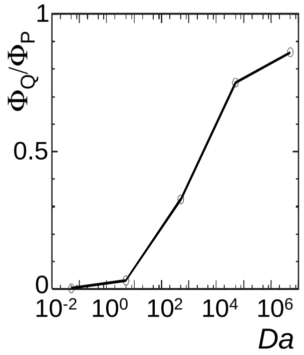

Figure 25. The product ratio Φ_Q/Φ_P as a function of the Damkohler number.

in the case of poor mixing (high Da) component *B* is not very well able to penetrate the reaction zone which in turn allows for the formation of *Q*. Figure 25 illustrates this point further. Here it is shown how the product ratio varies with Da between its theoretical limits (10^{-3} for Da→0; 1 for Da→∞).

Appendix: Coefficients in hydrodynamic force relations

Drag force:

$$C_{\mathrm{D}} = \frac{24}{\mathrm{Re}_{\mathrm{p}}}\left(1.0 + 0.15\,\mathrm{Re}_{\mathrm{p}}^{0.687}\right) \quad \mathrm{Re}_{\mathrm{p}} < 1{,}000$$

$$C_{\mathrm{D}} = 0.44 \quad \mathrm{Re}_{\mathrm{p}} \geq 1{,}000 \tag{A.1}$$

Saffman force:

$$C_{\mathrm{S}} = \frac{4.1126}{\mathrm{Re}_{\mathrm{S}}^{0.5}}\left[\left(1.0 - 0.234\left(\frac{\mathrm{Re}_{\mathrm{S}}}{\mathrm{Re}_{\mathrm{p}}}\right)^{0.5}\right)e^{-0.1\mathrm{Re}_{\mathrm{p}}} + 0.234\left(\frac{\mathrm{Re}_{\mathrm{S}}}{\mathrm{Re}_{\mathrm{p}}}\right)^{0.5}\right] \quad \mathrm{Re}_{\mathrm{p}} < 40$$

$$C_{\mathrm{S}} = 0.1524 \quad \mathrm{Re}_{\mathrm{p}} \geq 40 \tag{A.2}$$

Magnus force:

$$C_{\mathrm{M}} = 0.45 + \left(\frac{\mathrm{Re}_{\mathrm{R}}}{\mathrm{Re}_{\mathrm{p}}} - 0.45\right)e^{-0.05684\,\mathrm{Re}_{\mathrm{R}}^{0.4}\,\mathrm{Re}_{\mathrm{p}}^{0.3}} \tag{A.3}$$

References

C.K. Aidun, Y. Lu, and E.J. Ding (1998). Direct analysis of particle suspensions with inertia using the discrete Boltzmann equation. *Journal of Fluid Mechanics*, 373: 287-311.

K. Alvelius (1999). Random forcing of three-dimesnional homogeneous turbulence. *Physics of Fluids*, 11: 1880-1889.

P.L Bhatnagar, E.P. Gross and M. Krook (1954). A model for collision processes in gases. I: small amplitude processes in charged and neutral one-component system. *Physical Review*, 94: 511-525.

M. Chen, K. Kontomaris and J.B. McLaughlin (1998). Direct numerical simulation of droplet collisions in a turbulent channel flow. Part I: collision algorithm. *International Journal of Multiphase Flow*, 24: 1079-1103.

S. Chen, Z. Wang, X.W. Shan and G.D. Doolen (1992). Lattice-Boltzmann computational fluid dynamics in three dimensions. *Journal of Statistical Physics*, 68: 379-400.

S. Chen and G.D. Doolen (1998). Lattice-Boltzmann method for fluid flows. *Annual Review Fluid Mechanics*, 30: 329-364.

P.J. Colucci, F.A. Jaberi, P. Givi and S.B. Pope (1998). Filtered density functions for large eddy simulation of turbulent reacting flows. *Physics of Fluids*, 10: 499-515.

C. Crowe, M. Sommerfeld and Y. Tsuji (1998). *Multiphase flows with droplets and particles,* CRC Press, Boca Raton.

J.W. Deardorff (1973). The use of subgrid transport equations in a three-dimensional model of atmospheric turbulence. *Journal of Fluids Engineering*, 95: 429-438.

S.C.R. Dennis, S.N. Singh and D.B. Ingham (1980). The steady flow due to rotating sphere at low and moderate Reynolds numbers. *Journal of Fluid Mechanics*, 101: 257-279.

J.J. Derksen and H.E.A. van den Akker (1999). Large eddy simulations on the flow driven by a Rushton turbine. *AIChE Journal*, 45: 209-221.

J.J. Derksen (2003). Numerical simulation of solids suspension in a stirred tank. *AIChE Journal*, 49: 2700-2714.

J.J. Derksen (2005). Simulations of confined turbulent vortex flow. *Computers & Fluids*, 34:301-318.

E.J. Ding and C.K. Aidun (2003). Extension of the lattice-Boltzmann method for direct simulation of suspended particles near contact. *Journal of Computational Physics*, 112: 685-708.

P. Duru and E. Guazelli (2002). Experimental investigations on the secondary instability of liquid-fluidized beds and the formation of bubbles. *Journal of Fluid Mechanics*, 470: 359-382.

J.G.M. Eggels (1994). *Direct and large eddy simulation of turbulent flow in a cylindrical pipe geometry*. PhD Thesis, Delft University of Technology.

J.G.M. Eggels and J.A. Somers (1995). Numerical simulation of free convective flow using the lattice-Boltzmann scheme. *International Journal of Heat and Fluid Flow*, 16: 357-364.

S. Elghobashi (1994). On predicting particle-laden turbulent flows. *Applied Scientific Research*, 52: 309-329.

M.P. Escudier, J. Bornstein and N. Zehnder (1980). Observations and LDA measurements of confined turbulent vortex flow. *Journal of Fluid Mechanics*, 98: 49-63.

U. Frisch, B. Hasslacher and Y. Pomeau (1986). Lattice-gas automata for the Navier-Stokes equations. *Physical Review Letters*, 56: 1505-1508.

U. Frisch, D. d'Humieres, B. Hasslacher, Y. Pomeau and J.P. Rivet (1987). Lattice gas hydrodynamics in two and three dimensions. *Complex Systems*, 1: 649.

M. Germano, U. Piomelli, P. Moin and W. Cabot (1991). A dynamic subgrid-scale eddy-viscosity model. *Physics of Fluids A*, 3: 1760-1765.

D. Goldstein, R. Handler and L. Sirovich (1993) Modeling a no-slip flow boundary with an external force field. *Journal of Computational Physics*, 105: 354-366.

A. Harten (1983). High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 49: 357-393.

H. Hartmann, J.J. Derksen and H.E.A. van den Akker (2006). Numerical simulation of a dissolution process in a stirred tank reactor. *Chemical Engineering Science*, 61: 3025-3032.

H. Hasimoto (1959). On the periodic fundamental solutions of the Stokes equation and their application to viscous flow past a cubic array of spheres. *Journal of Fluid Mechanics*, 5: 317-328.

S. Kim and S.J. Karrila (1991). *Microhydrodynamics: Principles and selected applications*. Butterworth-Heinemann.

K. Kontomaris, T.J. Hanratty and J.B. McLaughlin (1992). An algorithm for tracking fluid particles in a spectral simulation of turbulent channel flow. *Journal of Computational Physics*, 103: 231-242.

A.J.C. Ladd (1994). Numerical simulations of particle suspensions via a discretized Boltzmann equation. Part I: Theoretical Foundation. *Journal of Fluid Mechanics*, 271: 285-309.

P. Lallemand and L.S. Luo (2000). Theory of the lattice-Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability. *Physical Review E*, 61: 6546-6562.

M. Lesieur and O. Métais (1996). New trends in large-eddy simulations of turbulence. *Annual Review Fluid Mechanics*, 28: 45-82.

P.J. Mason and N.S. Callen (1983). On the magnitude of the subgrid-scale eddy coefficient in large-eddy simulations of turbulent channel flow. *Journal of Fluid Mechanics*, 162: 439-462.

M.R. Maxey and J.J. Riley (1983). Equation of motion for a small rigid sphere in a nonuniform flow. *Physics of Fluids*, 26: 883-889.

R. Mei (1992). An approximate expression for the shear lift force on a spherical particle at finite Reynolds numbers. *International Journal of Multiphase Flow*, 18: 145-147.

N.-Q. Nguyen and A.J.C. Ladd (2002). Lubrication corrections for lattice-Boltzmann simulations of particle suspensions. *Physical Review, E* 66: 046708.

F. Odar and W.S. Hamilton (1964). Forces on a sphere accelerating in a viscous fluid. *Journal of Fluid Mechanics*, 18: 302-314.

B. Oesterlé and T. Bui Dinh (1998). Experiments on the lift of a spinning sphere in the range of intermediate Reynolds numbers. *Experiments in Fluids*, 25: 16-22.

W.E. Ranz and W.R. Marshall (1952). Evaporation from drops. *Chemical Engineering Progress*, 48: 141-146.

M. Rohde, J.J. Derksen and H.E.A. van den Akker (2002). Volumetric method for calculating the flow around a moving object in lattice-Boltzmann schemes. *Physical Review E*, 65: 056701.

P. Sagaut, P. Comte and F. Ducros (2000). Filtered subgrid-scale models. *Physics of Fluids*, 12: 233.

C.E. Schem and F.B. Lipps (1976). Some results from a simplified three-dimensional numerical model of atmospheric turbulence. *Journal of Atmospheric Science*, 33: 1021-1041.

H. Schmidt and U. Schumann (1989). Coherent structure of the convective boundary layer from large-eddy simulations. *Journal of Fluid Mechanics*, 200: 511-562.

J. Smagorinsky (1963). General circulation experiments with the primitive equations: 1. The basic experiment. *Monthly Weather Review*, 91: 99-164.

J.A. Somers (1993) Direct simulation of fluid flow with cellular automata and the lattice-Boltzmann equation. *Applied Scientific Research*, 51: 127-133.

M. Sommerfeld (2001). Validation of a stochastic Lagrangian modelling approach for inter-particle collisions in homogeneous isotropic turbulence. *International Journal of Multiphase Flow*, 27: 1829-1858.

S. Succi (2001). *The lattice Boltzmann equation for fluid dynamics and beyond*. Clarondon Press, Oxford.

A. ten Cate, C.H. Nieuwstad, J.J. Derksen and H.E.A. van den Akker (2002). PIV experiments and lattice-Boltzmann simulations on a single sphere settling under gravity. *Physics of Fluids*, 14: 4012-4025.

A. ten Cate, J.J. Derksen, L.M. Portela and H.E.A. van den Akker (2004). Fully resolved simulations of colliding spheres in forced isotropic turbulence. *Journal of Fluid Mechanics*, 519: 233-271.

A. ten Cate, E. van Vliet, J.J. Derksen, and H.E.A. van den Akker (2006). Application of spectral forcing in lattice-Boltzmann simulations of homogeneous turbulence. *Computers & Fluids,* 35:1239-1251.

H. Tennekes and J.L. Lumley (1972). A first course in turbulence. MIT Press, Cambridge MA.

E.R. van Driest (1965). On turbulent flow near a wall. *Journal of Aerodynamic Science*, 23: 1007-1011.

E. van Vliet, J.J. Derksen and H.E.A. van den Akker (2005). Turbulent mixing in a tubular reactor: assessment of an FDF/LES approach. *AIChE Journal*, 51: 725-739.

P.R. Voke (1996). Subgrid-scale modelling at low mesh Reynolds number. *Theoretical and Computational Fluid Dynamics*, 8: 131-143.

R. Weber, F. Boysan, W.H. Ayers and J. Swithenbank (1984). Simulation of dispersion of heavy particles in confined turbulent flows. *AIChE Journal,* 30: 490-492.

Y. Yamamoto, M. Potthoff, T. Tanaka, T. Kajishima Y. Tsuji (2001). Large-eddy simulation of turbulent gas-particle flow in a vertical channel: effect of considering inter-particle collisions. *Journal of Fluid Mechanics*, 442: 303-334.