Chapter 2

Liquid Dynamics

In this chapter fluid flow in stationary containers is studied. First, the mathematical model is explained in section 2.1. Then, in section 2.2, the representation of complex geometries on a Cartesian grid is explained. Section 2.3 discusses in detail the discretisation of the governing equations and boundary conditions. Methods for advecting a free surface are studied in section 2.4. Finally, in section 2.5, results are presented. For presentational reasons, the theory in this chapter is developed in two dimensions. In most situations this can be extended to three dimensions straightforwardly. If this is not the case, remarks on the three-dimensional extension are made.

2.1 Mathematical Model

In this section the mathematical model for the liquid dynamics is discussed: first the governing equations and thereafter the boundary conditions are stated.

2.1.1 Governing Equations

Throughout this thesis, the flow of a Newtonian, incompressible fluid with density ρ and molecular viscosity μ is studied. The motion of such a fluid is governed by an equation for conservation of mass

$$\nabla \cdot \boldsymbol{u} = 0, \tag{2.1}$$

and an equation for conservation of momentum

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\, \boldsymbol{u} = -\frac{1}{\rho} \left(\nabla p - (\nabla \cdot \mu \nabla)\, \boldsymbol{u}\right) + \boldsymbol{F}. \tag{2.2}$$

Equation (2.1) is referred to as the continuity equation, while equation (2.2) is better known as the Navier-Stokes equation (sometimes, the term Navier-Stokes equations refers to the combination of equations (2.1) and (2.2) — in this thesis the context makes clear whether by the term Navier-Stokes equations only the momentum equations or the combination of these with the continuity equation is meant). In these equations $\boldsymbol{u} = (u, v, w)^T$ is the velocity of the liquid and p denotes the liquid pressure. The velocity vector and pressure depend on three spatial variables and on time. The vector $\boldsymbol{F} = (F_x, F_y, F_z)^T$

represents acceleration due to an external force, *e.g.* gravity. Often, the left-hand side of equation (2.2) is abbreviated using the material derivative, *i.e.*

$$\frac{D\boldsymbol{u}}{Dt} = \frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\, \boldsymbol{u}.$$

The form in which the Navier-Stokes equations (2.1) and (2.2) are written, *i.e.* in differential form, is in particular suitable for a finite-difference method. In such a method the continuous derivatives are replaced by finite-difference formulas. Since in this thesis the Navier-Stokes equations are discretised using the (more physical) finite-volume method (see section 1.3), the Navier-Stokes equations are rewritten in conservation form. For an arbitrary control volume V (with boundary ∂V) in the liquid, conservation of mass changes to

$$\oint_{\partial V} \boldsymbol{u} \cdot \boldsymbol{n}\, dS = 0, \tag{2.3}$$

while conservation of momentum reads

$$\int_V \frac{\partial \boldsymbol{u}}{\partial t}\, dV + \oint_{\partial V} \boldsymbol{u}\boldsymbol{u}^T \cdot \boldsymbol{n}\, dS = -\frac{1}{\rho} \oint_{\partial V} (p\boldsymbol{I}_3 - \mu \nabla \boldsymbol{u}) \cdot \boldsymbol{n}\, dS + \int_V \boldsymbol{F}\, dV. \tag{2.4}$$

In these equations $\boldsymbol{n}$ is the outward-pointing normal on the boundary of the control volume V and $\boldsymbol{I}_3$ denotes the 3×3 identity matrix. Equations (2.3) and (2.4) are in the form that will be used for the discretisation; see section 2.3.

In the present study, in most cases, not the entire flow domain is filled with liquid; a free surface separates the liquid from the surrounding air. The location of this free surface is not known in advance and needs to be solved as part of the flow problem. If the free surface is described by an equation $S(x, y, z, t) = 0$, then the temporal evolution of the free surface satisfies

$$\frac{DS}{Dt} \equiv \frac{\partial S}{\partial t} + (\boldsymbol{u} \cdot \nabla)\, S = 0,$$

stating that it propagates with the liquid velocity.

Note that the Navier-Stokes equations are only solved in the liquid; the motion of the ambient air is neglected, which is justified by the large differences in molecular viscosity and density between air and liquid. However, in certain applications that are beyond the scope of this thesis, the dynamics of the surrounding air does play an important role, requiring a more extensive mathematical model than applicable here (see for example [57, 92]).

2.1.2 Boundary Conditions

Boundary conditions are needed at the solid boundary and at the free surface. Further, if surface tension can not be neglected (*e.g.* in a micro-gravity environment), a boundary condition is required at the contact line, *i.e.* the intersection of the free surface with the solid boundary.

Solid Boundary

At the solid boundary, in general, the no-slip boundary condition for a viscous fluid is applied, *i.e.*

$$\boldsymbol{u} = 0$$

is prescribed at the boundary of the flow domain, stating that the solid boundary is impermeable and that the fluid sticks to the wall due to viscous effects. However, in some situations a free-slip boundary condition

$$u_n = 0 \quad \text{and} \quad \frac{\partial u_t}{\partial \boldsymbol{n}} = 0$$

is prescribed, neglecting shear stress at the solid boundary. In the latter equation $u_n = \boldsymbol{u} \cdot \boldsymbol{n}$ and $u_t = \boldsymbol{u} \cdot \boldsymbol{t}$ denote the normal and tangential velocity at the solid boundary respectively. See also figure 2.1.

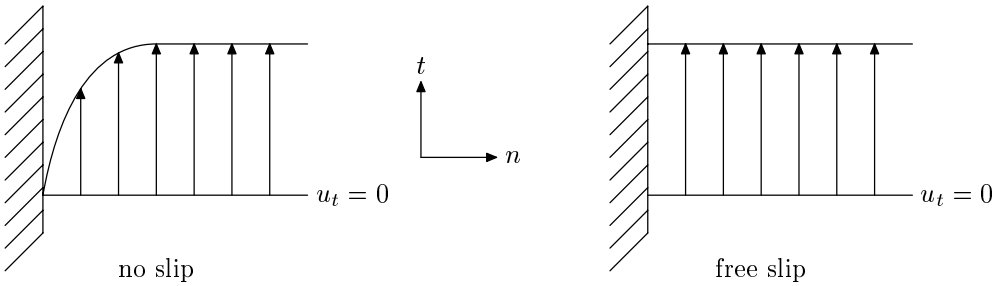

Figure 2.1: *Velocity profiles near the solid boundary for the no-slip and free-slip boundary condition.*

Free Surface

At the free surface, boundary conditions are needed both for the velocity and the pressure. Balancing the forces at the free surface results in a tangential free-surface condition

$$\mu \left(\frac{\partial u_n}{\partial \boldsymbol{t}} + \frac{\partial u_t}{\partial \boldsymbol{n}} \right) = 0, \tag{2.5}$$

which states that the air exerts no tangential stress on the fluid, and a normal condition

$$-p + 2\mu \frac{\partial u_n}{\partial \boldsymbol{n}} = -p_0 + \sigma\kappa. \tag{2.6}$$

In these equations u_n is the velocity normal to the free surface and u_t denotes the velocity in tangential direction. Further, p_0 is the ambient pressure of the air, σ is the surface tension of the fluid, and κ denotes the mean curvature of the free surface. If the free surface is described by $S(x, y, z, t) = 0$, then the mean curvature κ is given by

$$\kappa = \nabla \cdot \boldsymbol{n} = \nabla \cdot \left(\frac{\nabla S}{|\nabla S|} \right), \tag{2.7}$$

where $\boldsymbol{n}$ is the normal at the free surface.

For high values of the Reynolds number

$$Re = \frac{\rho U L}{\mu},$$

where U is a characteristic value of the velocity and L is a characteristic length scale, the second term on the left-hand side of equation (2.6) can be ignored since it is small compared to the other terms [36]. This simplification is made in this thesis also because of the small value of the viscosity that is used throughout this thesis. For high values of either the Bond number Bo or the Weber number We, which are defined by

$$Bo = \frac{\rho g L^2}{\sigma} \quad \text{and} \quad We = \frac{\rho L U^2}{\sigma},$$

respectively, also the second term on the right-hand side of (2.6) can be ignored. Indeed, for high Bo or We numbers, surface tension can be neglected since gravitational or inertial forces dominate. In this thesis, however, the emphasis will be on low values of the Bo and We number, whence the right-hand side of (2.6) can not be simplified.

Contact Angle

If surface tension can not be neglected (low values of the Bond number Bo and the Weber number We), *e.g.* for fluid flow in a micro-gravity environment, then the mean curvature of the free surface needs to be computed in order to apply boundary condition (2.6) at the free surface. For determining the mean curvature, spatial derivatives have to be computed (see equation (2.7)), whence boundary conditions are needed at the contact line where the free surface intersects the solid boundary. In this thesis these boundary conditions are supplied by a static contact angle θ, which is the angle between the normal at the free surface and the normal at the solid boundary. The contact angle is a quantity that depends on the material properties of liquid, air, and solid boundary.

For a schematic overview of the mathematical model see figure 2.2.

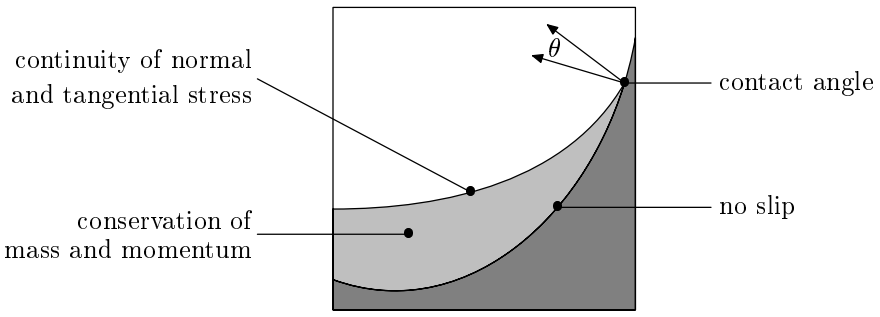

Figure 2.2: *Schematic overview of the mathematical model. Dark shading represents solid body and light shading denotes liquid.*

2.2 Geometry Discretisation

For a numerical method in computational fluid dynamics the choice of the grid is one of the first steps that has to be performed. Basically, a computational grid can be either structured or unstructured and is either boundary fitted or non-boundary fitted (see figure 2.3 and section 1.2). For complex geometries (in the figure the flow domain is

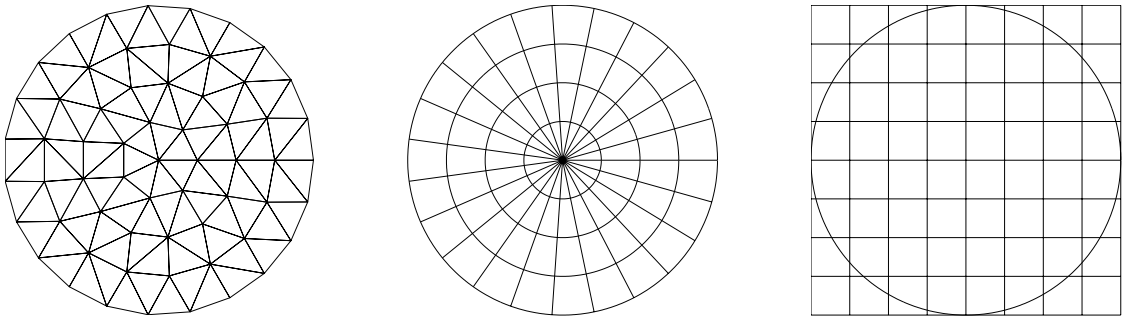

Figure 2.3: *Illustration of an unstructured, boundary-fitted grid (left), structured, boundary-fitted grid (middle), and structured, non-boundary-fitted grid (right) in the case that the flow domain is a circle. The latter is also called a Cartesian grid.*

a circle) the generation of an unstructured grid is relatively easy. However, the word *unstructured* already reveals the disadvantage of such a grid: the computational cells are not ordered and have different orientation and size. This makes bookkeeping a non-trivial problem, especially when a free surface needs to be tracked through the grid. In a structured grid the cells are ordered, but the generation of a structured, boundary-fitted grid is quite difficult and may be more time consuming than the flow simulation itself. A Cartesian grid, as shown in the right of figure 2.3, has two important advantages: the computing time for the generation of a Cartesian grid is negligible and already a lot of research has been done on free-surface tracking on orthogonal grids (see the references in chapter 1). However, if the flow domain is not staircase shaped, then its boundary is not aligned with grid lines; the boundary of the flow domain cuts the grid cells in various manners as can be seen in the right of figure 2.3. Hence, discretisation of the governing equations and the boundary conditions needs special care on a Cartesian grid. This is the approach that is chosen in this thesis.

2.2.1 Apertures

In order to recognise the flow domain on a Cartesian grid, a volume aperture F^b and edge apertures A^x, A^y, and A^z are introduced, indicating the fraction of a cell and cell face that is open for flow. See figure 2.4, where δx and δy denote the mesh size in x- and y-direction, for a graphical illustration of apertures. Note that in three dimensions the edge apertures contain information about the area of a cell face that is open for flow. Volume and edge apertures are used in the discretisation of the Navier-Stokes equations.

Apart from apertures for the geometry, one more volume aperture F^s is introduced for tracking the free surface. This aperture denotes the fractional volume of a cell that actually is occupied with fluid and is also known as the volume-of-fluid (VOF) function

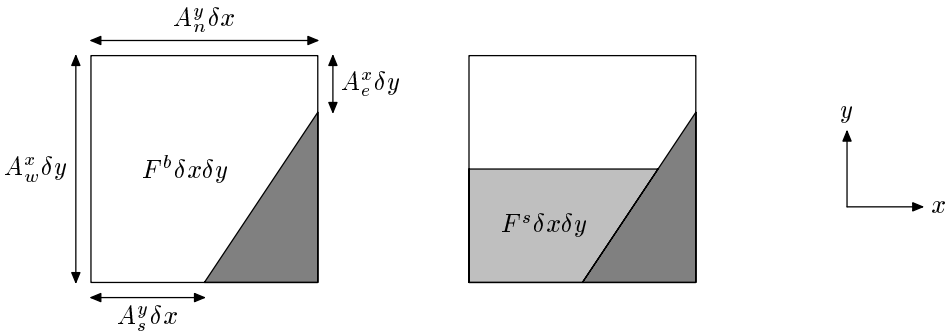

Figure 2.4: *Illustration of volume and edge apertures for the geometry (left) and volume aperture (or VOF function) for the free surface (right). The dark shading represents solid body, the lighter shading represents fluid.*

or colour function (see section 1.4). The volume apertures for the geometry and the free surface are related by $0 \leq F^s \leq F^b \leq 1$.

2.2.2 Labeling

Based on the apertures, every computational cell is given a label. First, all the interior cells containing no fluid, *i.e.* cells with $F^b > 0$ and $F^s = 0$, are labeled as empty cells (abbreviated with E). Non-empty interior cells adjacent to empty cells are labeled as surface cells (S). These cells contain part of the free surface and satisfy $F^b > 0$ and $0 < F^s \leq F^b$. All the remaining non-empty interior cells are called full cells (F). Note that these cells need not be full in the regular meaning of the word. The cells with $F^b = 0$ are flagged as boundary cells (B). In figure 2.5 an example of a label configuration is shown. A cell face is labeled based on the two computational cells it belongs to. Thus, for example, a cell face between an F cell and a B cell is called an FB cell face.

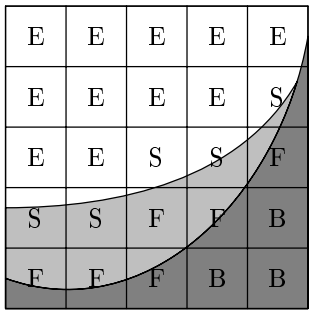

Figure 2.5: *Cell labels.*

2.3 Discretisation of Governing Equations

For the discretisation of the continuity equation and the Navier-Stokes equations, the finite-volume method is applied in such a way that the discrete difference operators for

convection and diffusion preserve the symmetry properties of the corresponding continuous differential operators [81]. Thus, convection is discretised with a skew-symmetric difference operator and diffusion is approximated by a symmetric, negative-definite difference operator. Further, the coefficient matrix of the pressure gradient is equal to minus the transpose of the discrete divergence operator. On the one hand this fulfils the desire to let the discrete operators inherit spectral properties of the continuous operators, on the other hand this approach assures a stable integration of the Navier-Stokes equations on any grid and guarantees conservation of mass, momentum, and, in the absence of viscosity and external forces, kinetic energy as is shown later in this section.

2.3.1 Continuity Equation

Discretisation of the continuity equation (2.3) is rather straightforward. Consider a computational cell, as shown in figure 2.6, of which some part is occupied by the solid body (the shaded area). The velocities are staggered in this cell, meaning that a horizontal

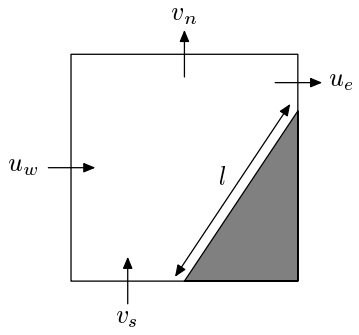

Figure 2.6: *Conservation cell for discretisation of the continuity equation.*

velocity (u) is placed at a vertical cell face and a vertical velocity (v) is placed at a horizontal cell face [29]. Note that such a velocity need not be positioned in the centre of that cell face; along the entire cell face (at least that part which is open for flow) this velocity holds. As a control volume V for discretisation of the continuity equation the part of the cell that is open for flow is chosen. Discrete conservation of mass implies that the sum of all the mass fluxes (velocity multiplied by area) through the boundary ∂V of V should vanish. For the cell in figure 2.6 this results in

$$u_e A_e^x \delta y + v_n A_n^y \delta x - u_w A_w^x \delta y - v_s A_s^y \delta x + 0 \cdot l = 0, \tag{2.8}$$

where some notation of figure 2.4 has been used (see also [1]). The last term on the left-hand side of this equation represents the mass flux through the solid boundary, which, of course, equals zero. Since the mass flux through a solid boundary always vanishes in the continuity equation, equation (2.8) can be applied in every cell independent of the configuration of the solid boundary. If a cell and its faces do not contain part of the solid body, then all the edge apertures are equal to one, whence equation (2.8) is identical to a second-order central discretisation of equation (2.1).

2.3.2 Navier-Stokes Equations

In this section the spatial discretisation of the momentum equations (2.4) is discussed. Only the momentum equation in x-direction is considered; the other directions follow a similar approach. First the control volumes are defined, thereafter the spatial discretisation of the time derivative, the convective and diffusive terms, the pressure term, and acceleration due to external forces is discussed. Finally, an overview of the spatial discretisation is given and the relation between the discrete divergence and gradient operators is discussed. The temporal discretisation of the momentum equations is explained in section 2.3.4.

Control Volumes

For uncut cells, the control volume of a horizontal velocity $u_{i,j}$, located on a vertical cell face $x = x_i$ and $y_{j-1} \leq y \leq y_j$, is usually taken as illustrated in figure 2.7. The control

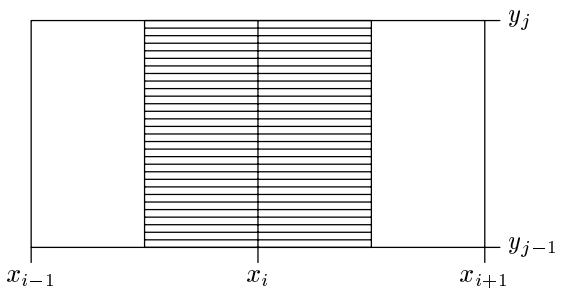

Figure 2.7: *Conservation cell (indicated with horizontal lines) for the momentum equation in the case of uncut cells.*

volume, indicated with horizontal lines, consists of half of the left-hand cell $[x_{i-1}, x_i] \times [y_{j-1}, y_j]$ and half of the right-hand cell $[x_i, x_{i+1}] \times [y_{j-1}, y_j]$ (in later references these cells are denoted with indices (i, j) and $(i + 1, j)$ respectively). For defining control volumes in the case of cut cells, a closer look at figure 2.7 is required. The part of the control volume that is located in cell $(i + 1, j)$, for example, may be regarded as the union of an infinite number of horizontal line segments, running from $x = x_i$ to $x = x_{i+1}$, which are bisected; the half nearest to grid line $x = x_i$ is taken as part of the control volume.

If the solid boundary intersects cell (i, j) or $(i + 1, j)$ (or both), then the control volume for the velocity $u_{i,j}$ is adapted, but basically the same procedure is used. Consider, again, cell $(i + 1, j)$. Horizontal line segments are drawn either from $x = x_i$ to $x = x_{i+1}$, from $x = x_i$ to the solid boundary, or from the solid boundary to $x = x_{i+1}$ (depending on the location of the boundary). These line segments are bisected and the half nearest to grid line $x = x_i$ is chosen as part of the control volume [82]. This procedure is illustrated by means of two examples in figure 2.8 (in this figure the dark shading denotes the part of the computational cells that is not open for flow). Note that also for uncut cells, because of construction, this procedure leads to a control volume for $u_{i,j}$ that bisects the "open" part of the cells (i, j) and $(i + 1, j)$. In three dimensions exactly the same procedure is used for constructing control volumes.

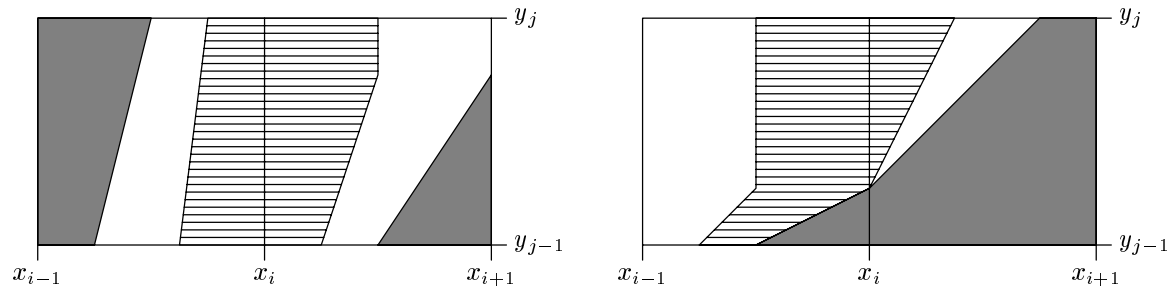

Figure 2.8: *Conservation cell (horizontal lines) for the momentum equation for two different configurations of the solid boundary.*

Time Derivative

The time derivative in the Navier-Stokes equations is discretised in space (the temporal discretisation is discussed in section 2.3.4) using the midpoint rule, *i.e.*

$$\int_V \frac{\partial u}{\partial t}\, dV \doteq \frac{\partial u_c}{\partial t} F_c^b \delta x_c \delta y, \tag{2.9}$$

where V is the control volume corresponding to the horizontal velocity u_c. This control volume has a volume of $F_c^b \delta x_c \delta y$, where $F_c^b = \left(F_w^b \delta x_w \delta y + F_e^b \delta x_e \delta y\right) / \left(\delta x_w \delta y + \delta x_e \delta y\right)$ and $\delta x_c = \frac{1}{2}\left(\delta x_w + \delta x_e\right)$ (F_w^b and F_e^b denote the volume apertures of the western and eastern computational cell contributing to the control volume of u_c; see also figure 2.9 for explanation of the notation).

Convective Terms

In the momentum equation in x-direction the convective terms (in conservative form) read

$$\oint_{\partial V} u\boldsymbol{u} \cdot \boldsymbol{n}\, dS. \tag{2.10}$$

Note the difference between the scalar u and the vector $\boldsymbol{u}$ in this expression. The scalar velocity is the horizontal momentum that is advected with a velocity equal to the vector velocity. The discretisation of (2.10) is best illustrated with an example. Consider hereto the situation in figure 2.9. The boundary ∂V of the control volume is indicated with a dashed line and consists of seven line segments, numbered from 1 to 7. On each of these segments the integrand in expression (2.10) is discretised as a product of the horizontal velocity (u) and the mass flux through this segment ($\boldsymbol{u} \cdot \boldsymbol{n}\, dS$), *i.e.*

$$\oint_{\partial V} u\boldsymbol{u} \cdot \boldsymbol{n}\, dS \doteq \sum_{k=1}^{7} u_k f_k,$$

where u_k and f_k denote the horizontal velocity at segment k and the mass flux through segment k respectively (the mass flux is positive if liquid leaves the control volume). The velocity u_1 at segment 1 is defined as the average of a zero velocity on the right and a

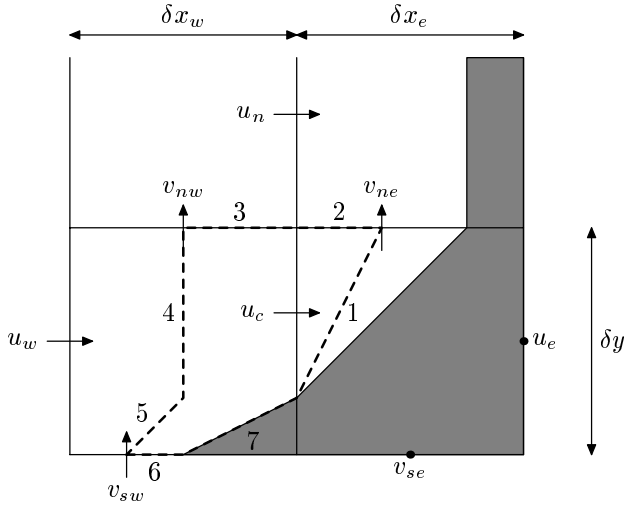

Figure 2.9: *Notation used in the discretisation of the convective and diffusive terms.*

velocity u_c on the left of segment 1. Note that, although the velocity u_c is drawn on the central cell face in figure 2.9, this velocity is characteristic throughout the entire control volume. Similarly, the zero velocity on the right of segment 1 holds for the total volume on the right of this segment and not only at the solid boundary. The mass flux through segment 1 is defined as the average of a zero mass flux through the solid boundary and the mass flux through the central cell face, *i.e.* $u_c A_c^x \delta y$ (see figure 2.4 for notation of the apertures), where A_c^x is the edge aperture of the cell face corresponding to u_c. This seems a complicated definition since it depends on the location of the solid boundary. However, the formula

$$u_1 f_1 = \frac{1}{2}\left(u_e + u_c\right)\frac{1}{2}\left(u_e A_e^x \delta y + u_c A_c^x \delta y\right),$$

where u_e is a velocity defined at the eastern cell face (in this case, since the edge aperture A_e^x is zero, $u_e = 0$), can be applied at every eastern segment of a control volume; the edge apertures contain sufficient information to account for the presence of a solid body. The velocities u_2 and u_3 at segments 2 and 3 are defined as the average of the velocities u_n and u_c. Here the average is unweighted, even if the grid is stretched in y-direction, such that the resulting coefficient matrix for convection is skew symmetric. Indeed, if a weighting factor α would be used in the averaging of u_n and u_c, then the northern coefficient for u_c is equal to the sum of mass fluxes through segments 2 and 3 multiplied by α, while the southern coefficient for u_n is the same sum of mass fluxes with opposite sign multiplied by $1-\alpha$, whence the matrix for convection is skew symmetric if and only if $\alpha = \frac{1}{2}$. The mass fluxes through segments 2 and 3 are computed as $v_{ne}\frac{1}{2}A_{ne}^y \delta x_e$ and $v_{nw}\frac{1}{2}A_{nw}^y \delta x_w$ respectively. Thus, summarised for segments 2 and 3

$$u_2 f_2 + u_3 f_3 = \frac{1}{2}\left(u_n + u_c\right)\left(v_{ne}\frac{1}{2}A_{ne}^y \delta x_e + v_{nw}\frac{1}{2}A_{nw}^y \delta x_w\right).$$

For segments 4 and 5 the same approach is used as for segment 1, which results in

$$u_4 f_4 + u_5 f_5 = -\frac{1}{2}\left(u_w + u_c\right)\frac{1}{2}\left(u_w A_w^x \delta y + u_c A_c^x \delta y\right).$$

For both these segments the horizontal velocity is defined as the average of u_w on the left and u_c on the right. From this equation the separate mass fluxes through segments 4 and 5 are not clearly visible. However, since $A^x_w = 1$ in the example under consideration, these mass fluxes can be defined as $f_4 = -\frac{1}{2}\left(u_w A^x_c \delta y + u_c A^x_c \delta y\right)$ and $f_5 = -\frac{1}{2}\left(u_w \left(1 - A^x_c\right) \delta y + 0\right)$ respectively. Note that this formulation for segments 4 and 5 is similar to the one for segment 1: the eastern velocity and edge aperture have been replaced by their western equivalents and a minus sign has been added. Nevertheless, the situation with respect to the solid boundary is completely different in the eastern and western cell. For segment 6 a similar equation as for segments 2 and 3 is used. For segment 7, clearly, the mass flux is equal to zero. This makes the value for u_7 irrelevant. The convective contribution of segments 6 and 7 is written in a single formula for the southern part of the control volume, namely

$$u_6 f_6 + u_7 f_7 = -\frac{1}{2}\left(u_s + u_c\right)\left(v_{sw}\frac{1}{2}A^y_{sw}\delta x_w + v_{se}\frac{1}{2}A^y_{se}\delta x_e\right).$$

The terms of the discretisation along the seven segments are rearranged, such that the distinction between the unknown horizontal velocities and the coefficient matrix for convection (the mass fluxes) becomes more visible. This results in the following discrete version of expression (2.10)

$$\begin{aligned}
&\frac{1}{4}\left(u_e A^x_e \delta y + v_{ne} A^y_{ne} \delta x_e + v_{nw} A^y_{nw} \delta x_w - u_w A^x_w \delta y - v_{sw} A^y_{sw} \delta x_w - v_{se} A^y_{se} \delta x_e\right) u_c \\
&\qquad + \frac{1}{4}\left(u_e A^x_e \delta y + u_c A^x_c \delta y\right) u_e + \frac{1}{4}\left(v_{ne} A^y_{ne} \delta x_e + v_{nw} A^y_{nw} \delta x_w\right) u_n \\
&\qquad - \frac{1}{4}\left(u_w A^x_w \delta y + u_c A^x_c \delta y\right) u_w - \frac{1}{4}\left(v_{sw} A^y_{sw} \delta x_w + v_{se} A^y_{se} \delta x_e\right) u_s .
\end{aligned} \tag{2.11}$$

The coefficient of u_c, the diagonal entry of the coefficient matrix, vanishes since it represents the net mass flow through the boundaries of the eastern and western cell (this is equal to zero because of conservation of mass in F and S cells). Further, it is clear that the off-diagonal entries are skew symmetric. Thus, the discrete coefficient matrix for convection inherits this symmetry property of the continuous convective operator.

Diffusive Terms

Discretisation of the diffusive terms

$$\oint_{\partial V} \nabla u \cdot \boldsymbol{n}\, dS, \tag{2.12}$$

where the constant μ/ρ has been omitted, is more difficult than the convective terms since derivatives of the horizontal velocity are needed at the boundary of the control volume. Moreover, since the diffusive flux through the solid boundary need not be zero, a discretisation independent of the exact location of the solid boundary is not possible anymore. In figure 2.9, for example, the convective flux through segments 6 and 7 could be discretised together since the mass flux vanishes through a solid boundary. For the diffusive flux segments 6 and 7 have to be treated separately. In fact, segments of the

boundary of the control volume which coincide with the solid boundary require a different approach than segments located in the interior of the flow domain as is demonstrated in this section.

First, the general idea of the discretisation of expression (2.12) is explained. Hereto, note that the integrand in this expression can be written as $\partial u/\partial \boldsymbol{n}$. This formulation motivates the following discretisation of (2.12) for the control volume in figure 2.9

$$\oint_{\partial V} \nabla u \cdot \boldsymbol{n}\, dS \doteq \sum_{k=1}^{7} \frac{u_k - u_c}{|\boldsymbol{n}_k|} \oint_k dS,$$

where u_k is a velocity that is characteristic on the side of segment k in the direction of the outward-pointing normal. Further, $|\boldsymbol{n}_k|$ is the distance between u_k and u_c. However, in the finite-volume method, a velocity does not have a fixed position but instead has a constant value throughout its control volume. Hence, the distance between two velocities has to be approximated. Since $|\boldsymbol{n}_k|$ is a geometric quantity having the dimension of length, it is approximated by

$$|\boldsymbol{n}_k| = \frac{V_k}{A_k} \quad \text{with} \quad A_k = \oint_k dS, \tag{2.13}$$

where V_k is a volume corresponding to segment k of the control volume and A_k is the area of segment k. The precise choice of V_k varies per segment and is explained below.

Now, for the seven segments in figure 2.9 the discretisation of the diffusive terms is discussed in more detail. Hereto, the volume of the control volume is decomposed in two volumes, namely V_e and V_w, representing the parts of the control volume located in the eastern and western cell respectively. Using volume apertures it follows that $V_e = \frac{1}{2}F_e^b \delta x_e \delta y$ and $V_w = \frac{1}{2}F_w^b \delta x_w \delta y$. For segment 1 the volume V_1 is set to the volume of the eastern cell, *i.e.* $V_1 = 2V_e$, and the velocity u_1 is equal to u_e (which is zero in this particular example since $A_e^x = 0$). Thus, for the eastern part of the control volume the contribution to the diffusive flux equals

$$\frac{u_1 - u_c}{|\boldsymbol{n}_1|} A_1 = \frac{u_e - u_c}{2V_e} A_e^2,$$

where $A_e = A_1$ is the area of the eastern part of the control volume. The velocities u_2 and u_3 are equal to u_n. The volume V_2 is computed as $\frac{1}{2}V_{ne} + \frac{1}{2}V_e$, where V_{ne} is the volume of the part of the control volume for u_n that is located in the north-eastern cell, *i.e.* $V_{ne} = \frac{1}{2}F_{ne}^b \delta x_e \delta y_n$. Similarly, if $V_{nw} = \frac{1}{2}F_{nw}^b \delta x_w \delta y_n$, then $V_3 = \frac{1}{2}V_{nw} + \frac{1}{2}V_w$. Hence, for the northern part

$$\frac{u_2 - u_c}{|\boldsymbol{n}_2|} A_2 + \frac{u_3 - u_c}{|\boldsymbol{n}_3|} A_3 = \frac{u_n - u_c}{\frac{1}{2}V_{ne} + \frac{1}{2}V_e} \left(\frac{1}{2}A_{ne}^y \delta x_e\right)^2 + \frac{u_n - u_c}{\frac{1}{2}V_{nw} + \frac{1}{2}V_w} \left(\frac{1}{2}A_{nw}^y \delta x_w\right)^2.$$

The terms between brackets represent the area of segments 2 and 3. Segments 4 and 5 can be treated simultaneously since these are located in the same computational cell, which is expressed by the following definition

$$\frac{u_4 - u_c}{|\boldsymbol{n}_4|} A_4 + \frac{u_5 - u_c}{|\boldsymbol{n}_5|} A_5 \equiv \frac{u_{4,5} - u_c}{|\boldsymbol{n}_{4,5}|} A_{4,5}.$$

The velocity $u_{4,5}$ is set to u_w and the volume $V_{4,5}$ is set to $2V_w$, whence the western contribution becomes

$$\frac{u_{4,5}-u_c}{|\boldsymbol{n}_{4,5}|}A_{4,5} = \frac{u_w-u_c}{2V_w}A_w^2,$$

where A_w is the area of the western part of the control volume. Note the similarity between the eastern and western expressions. For the southern part of the control volume the same approach is used as for the northern part, *i.e.*

$$\frac{u_6-u_c}{|\boldsymbol{n}_6|}A_6 = \frac{u_s-u_c}{\frac{1}{2}V_{sw}+\frac{1}{2}V_w}\left(\frac{1}{2}A_{sw}^y\delta x_w\right)^2 + \frac{u_s-u_c}{\frac{1}{2}V_{se}+\frac{1}{2}V_e}\left(\frac{1}{2}A_{se}^y\delta x_e\right)^2.$$

The last term in this equation is somewhat unexpected since the southern part consists of segment 6 only (segment 7 is treated separately as shown below). However, this term is added to acquire a formula that can be applied in every cell. In fact, in the example in figure 2.9, this term does not contribute to the diffusive terms since $A_{se}^y = 0$. The last segment ($k = 7$) is different from the previous six segments because it coincides with the solid boundary. Since a velocity is needed on both sides of the segment, the part of the control volume that is located in the western cell (with volume V_w) is reflected about this segment. In this mirrored volume a velocity $-u_c$ (because of the no-slip boundary condition) is assumed. See figure 2.10 for a graphical illustration. The volume V_7 is taken

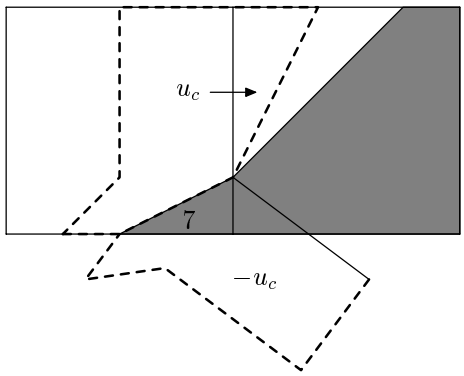

Figure 2.10: *Mirrored control volume for computing the diffusive flux through the solid boundary (segment 7).*

equal to V_w or, since the volume of the western control volume is equal to the mirrored volume,

$$V_7 = \frac{1}{2}V_w + \frac{1}{2}V_w,$$

i.e. the sum of half the control volumes on both sides of the segment. This trivial decomposition is made to show consistency with the treatment of segments 2, 3, and 6. So, for the last segment an extra contribution to the diffusive terms arises, namely

$$\frac{u_7-u_c}{|\boldsymbol{n}_7|}A_7 = \frac{-u_c-u_c}{\frac{1}{2}V_w+\frac{1}{2}V_w}A_{\text{bnd}}^2 = \frac{-2u_c}{V_w}A_{\text{bnd}}^2,$$

where A_{bnd} is the area of the solid boundary that is located in the western cell.

After rearranging terms the discretised diffusive terms for the control volume in figure 2.9 read

$$
\begin{aligned}
-\frac{1}{2}\left(\frac{A_e^2}{V_e}+\frac{\left(A_{ne}^y\delta x_e\right)^2}{V_{ne}+V_e}+\frac{\left(A_{nw}^y\delta x_w\right)^2}{V_{nw}+V_w}+\frac{A_w^2}{V_w}+\frac{\left(A_{sw}^y\delta x_w\right)^2}{V_{sw}+V_w}+\frac{\left(A_{se}^y\delta x_e\right)^2}{V_{se}+V_e}\right)u_c-\frac{2A_{\mathrm{bnd}}^2}{V_w}u_c \\
+\frac{1}{2}\frac{A_e^2}{V_e}u_e+\frac{1}{2}\left(\frac{\left(A_{ne}^y\delta x_e\right)^2}{V_{ne}+V_e}+\frac{\left(A_{nw}^y\delta x_w\right)^2}{V_{nw}+V_w}\right)u_n \\
+\frac{1}{2}\frac{A_w^2}{V_w}u_w+\frac{1}{2}\left(\frac{\left(A_{sw}^y\delta x_w\right)^2}{V_{sw}+V_w}+\frac{\left(A_{se}^y\delta x_e\right)^2}{V_{se}+V_e}\right)u_s.
\end{aligned}
\tag{2.14}
$$

From this formulation it is clear that the coefficient matrix for diffusion is symmetric, corresponding to the symmetry of the continuous diffusive operator. Further, the diagonal entry is, apart from a minus sign, the sum of the off-diagonal entries added with possible terms due to the solid boundary. Moreover, the diagonal entries are negative and the off-diagonal entries are positive. Thus, the discrete diffusive operator is a negative-definite matrix; all its eigenvalues are real and less than zero.

Pressure Term

Basically, the discretisation of the pressure term, which (without the factor $-1/\rho$) in the momentum equation in x-direction reads

$$
\oint_{\partial V} p n_x \, dS, \tag{2.15}
$$

where n_x is the first component of the outward-pointing normal vector $\boldsymbol{n}$ on the boundary ∂V of the control volume, follows the same approach as has been used for the convective and diffusive terms; the integrand is evaluated at the various segments of ∂V. However, since the pressure is an unstaggered quantity and only the first component of the normal vector appears in the integrand, the exact procedure is slightly different. Using figure 2.11, the discretisation of the pressure term is written as

$$
\oint_{\partial V} p n_x \, dS \doteq \sum_{k=1}^{7} p_k \oint_k n_x \, dS,
$$

where, again, the summation is over the seven segments of the control volume. The pressure along segment k is denoted by p_k; thus $p_1 = p_e$ and $p_4 = p_5 = p_7 = p_w$. Since the first component n_x of the normal vector is zero along segments 2, 3, and 6, these segments do not contribute to the sum. Regarding the other segments, denote the angle between segment k and the vertical by α_k (with $0 \leq \alpha_k < \frac{\pi}{2}$). Then

$$
\oint_k n_x \, dS = s_k \oint_k \cos\alpha_k \, dS,
$$

where $s_k = +1$ for $k = 1, 7$ and $s_k = -1$ for $k = 4, 5$. The integral in the right-hand side of this equation is equal to the vertical length of segment k. Summarised, the pressure

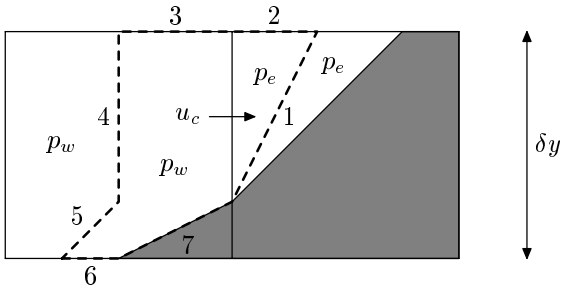

Figure 2.11: *Notation used in the discretisation of the pressure term.*

term is discretised as

$$\oint_{\partial V} p n_x \, dS \doteq p_e A_c^x \delta y - p_w A_c^x \delta y - p_w \left(1 - A_c^x\right) \delta y + p_w \left(1 - A_c^x\right) \delta y.$$

The contributions of segments 5 and 7 cancel each other since these segments lie in the same computational cell. This results in a formulation that, again, is independent of the exact location of the solid boundary, namely

$$\oint_{\partial V} p n_x \, dS \doteq \left(p_e - p_w\right) A_c^x \delta y. \tag{2.16}$$

External Force

An acceleration $\boldsymbol{F} = (F_x, F_y, F_z)^T$ due to external forces contributes to the momentum equation in x-direction as

$$\int_V F_x \, dV.$$

This term is discretised similar to the discretisation of the time derivative, *i.e.*

$$\int_V F_x \, dV \doteq F_{x_c} F_c^b \delta x_c \delta y_c, \tag{2.17}$$

where $F_c^b \delta x_c \delta y_c$ is the volume of the control volume V of the momentum equation under consideration and F_{x_c} is the value of F_x in this control volume (the acceleration may vary in space). If gravity is the only external force present, *i.e.* $\boldsymbol{F} = \boldsymbol{g} = (g_x, g_y, g_z)^T$ (constant in space), then this contribution to the momentum equations can be written as a boundary integral

$$\int_V F_x \, dV = \int_V \nabla \cdot \begin{pmatrix} g_x x \\ 0 \\ 0 \end{pmatrix} dV = \oint_{\partial V} \begin{pmatrix} g_x x \\ 0 \\ 0 \end{pmatrix} \cdot \boldsymbol{n} \, dS = \oint_{\partial V} g_x x n_x \, dS.$$

This boundary integral is similar to the pressure term (2.15) except for the factor x in the integrand. For segments 1, 4, 5, and 7 of the control volume in figure 2.11, the value of

x must be defined (the other segments do not contribute to the acceleration since $n_x = 0$ along these segments). In general, this value changes along a segment of the control volume. However, to be consistent with the pressure term, the value of x is kept constant along each segment and along segments sharing the same computational cell. With this in mind, the discretisation of the acceleration due to gravity becomes

$$\oint_{\partial V} g_x x n_x \, dS \doteq g_x x_e A_c^x \delta y - g_x x_w A_c^x \delta y = g_x \left(x_e - x_w\right) A_c^x \delta y, \tag{2.18}$$

where x_e and x_w are the (constant) x-coordinates along segment 1 and segments 4 and 5 respectively. In the case of uncut cells, the distance between x_e and x_w is equal to $\delta x_c = \frac{1}{2}\left(\delta x_e + \delta x_w\right)$. To be consistent with this situation, also for cut cells this choice is made.

Overview of the Spatial Discretisation

In this section the spatial discretisation of the continuity equation and the Navier-Stokes equation has been discussed. Before these equations are discretised in time in section 2.3.4, some notation is introduced and the evolution of kinetic energy is discussed.

Denote the vector of all (discrete) velocities that are solved from the momentum equations by $\boldsymbol{u}_h$ (thus, in two dimensions, this vector consists of velocity components u and v). The way in which the velocities are ordered is not important. Then, the (spatially) discretised continuity equations (in every cell, where the pressure has to be solved, one continuity equation is discretised) are written as

$$\mathcal{M}\boldsymbol{u}_h = 0. \tag{2.19}$$

Every row of the coefficient matrix $\mathcal{M}$ corresponds to the discrete continuity equation in a certain computational cell (the ordering of the computational cells is irrelevant). For example, the number of nonzero elements in the row corresponding to the eastern and western cell in figure 2.11 is equal to 2 and 4 respectively. Note that every entry in the matrix $\mathcal{M}$ contains geometrical information only, namely the area of a cell face that is open for flow (see also equation (2.8)). Similarly, the (spatially) discretised momentum equations are written as

$$\mathcal{V}\frac{d\boldsymbol{u}_h}{dt} + \mathcal{C}\left(\boldsymbol{u}_h\right)\boldsymbol{u}_h = -\frac{1}{\rho}\left(\mathcal{P}\boldsymbol{p}_h - \mu\mathcal{D}\boldsymbol{u}_h\right) + \boldsymbol{F}_h,$$

where $\boldsymbol{p}_h$ is a vector containing all the unknown pressures. The ordering of this vector must correspond to the ordering of the continuity equations, *i.e.* the j-th component of $\boldsymbol{p}_h$ contains the pressure in the cell of which the discrete continuity equation is found in the j-th row of matrix $\mathcal{M}$. The coefficient matrix $\mathcal{V}$ in front of the time derivative is a diagonal matrix. The entries in this matrix represent the volume of the control volumes (see equation (2.9)). Further, $\mathcal{C}$ and $\mathcal{D}$ are the coefficient matrices of the discretised convective and diffusive operators. The entries of matrix $\mathcal{C}$ are mass fluxes through cell faces (see expression (2.11)). Thus, apart from geometrical information, this matrix contains also velocities. The matrix $\mathcal{D}$ contains only geometrical information (see expression (2.14)). The vector $\boldsymbol{F}_h$ contains the discretisation of the acceleration due to external forces (see

equation (2.17) or (2.18)). The coefficient matrix $\mathcal{P}$ in front of the pressure contains the same geometrical information as $\mathcal{M}$ (see equation (2.16)).

Analytically, the divergence operator $(\nabla\cdot)$ and gradient operator (∇) are related by $\nabla = -(\nabla\cdot)^T$. For the discretisation method used in this thesis, this property is preserved, *i.e.* $\mathcal{P} = -\mathcal{M}^T$. This is explained using figure 2.12. Assume that the vector $\boldsymbol{p}_h$ is ordered

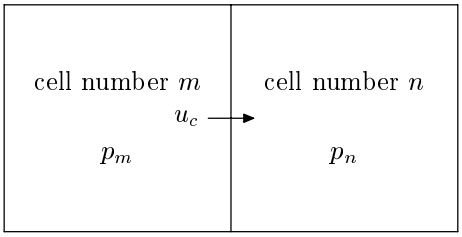

Figure 2.12: *Notation used in demonstrating the relation between the discrete divergence and gradient operator.*

such that the pressures p_m and p_n appear as the m-th and n-th entry respectively. In this case, the discrete continuity equations in cells m and n have entries in the matrix $\mathcal{M}$ that operate on the velocity u_c. If only these entries (located in rows m and n) in the matrix $\mathcal{M}$ are shown, then the discrete continuity equation (see also equation (2.8)) has the following form

$$\mathcal{M}\boldsymbol{u}_h = \begin{pmatrix} & \vdots & \\ \cdots & A_c^x \delta y & \cdots \\ & \vdots & \\ \cdots & -A_c^x \delta y & \cdots \\ & \vdots & \end{pmatrix} \begin{pmatrix} \vdots \\ u_c \\ \vdots \end{pmatrix} = 0.$$

In the discrete momentum equation for u_c the pressures in cell m and n appear. This means that in the row corresponding to the momentum equation for u_c, columns m and n of the matrix $\mathcal{P}$ contain nonzero elements (see also equation (2.16)), *i.e.*

$$\mathcal{P}\boldsymbol{p}_h = \begin{pmatrix} & \vdots & & \vdots & \\ \cdots & -A_c^x \delta y & \cdots & A_c^x \delta y & \cdots \\ & \vdots & & \vdots & \end{pmatrix} \begin{pmatrix} \vdots \\ p_m \\ \vdots \\ p_n \\ \vdots \end{pmatrix}.$$

From this it follows that indeed $\mathcal{P} = -\mathcal{M}^T$, whence the discretised Navier-Stokes equations can be written as

$$\mathcal{V}\frac{d\boldsymbol{u}_h}{dt} + \mathcal{C}\left(\boldsymbol{u}_h\right)\boldsymbol{u}_h = -\frac{1}{\rho}\left(-\mathcal{M}^T\boldsymbol{p}_h - \mu\mathcal{D}\boldsymbol{u}_h\right) + \boldsymbol{F}_h. \tag{2.20}$$

An attractive feature of the spatial discretisation that has been presented in this section is that it reduces to a standard second-order finite-difference discretisation when it is applied to a staircase-shaped geometry on a uniform grid.

2.3.3 Evolution of Kinetic Energy

With respect to the stability of the spatial discretisation it is interesting to see how discrete energy evolves in time. Hereto, first the kinetic energy is discretised in space as

$$\frac{1}{2}\int \rho|\boldsymbol{u}|^2\,dV \doteq \frac{1}{2}\left\langle \boldsymbol{u}_h, \rho\mathcal{V}\boldsymbol{u}_h\right\rangle \equiv E_h,$$

where the integral is over the entire flow domain and $\langle\cdot,\cdot\rangle$ denotes the standard inner product for a finite-dimensional vector space. Hence, the evolution of discrete energy is given by

$$\frac{dE_h}{dt} = \frac{1}{2}\left\langle \frac{d\boldsymbol{u}_h}{dt}, \rho\mathcal{V}\boldsymbol{u}_h\right\rangle + \frac{1}{2}\left\langle \boldsymbol{u}_h, \rho\mathcal{V}\frac{d\boldsymbol{u}_h}{dt}\right\rangle = \left\langle \boldsymbol{u}_h, \rho\mathcal{V}\frac{d\boldsymbol{u}_h}{dt}\right\rangle,$$

where the last equality is justified since $\boldsymbol{u}_h$ is a real vector and $\mathcal{V}$ is a real, diagonal matrix. Using the discretised momentum equations (2.20), this is equal to

$$\begin{aligned}\frac{dE_h}{dt} &= \left\langle \boldsymbol{u}_h, -\rho\mathcal{C}\boldsymbol{u}_h + \mathcal{M}^T\boldsymbol{p}_h + \mu\mathcal{D}\boldsymbol{u}_h + \rho\boldsymbol{F}_h\right\rangle \\ &= -\rho\left\langle \boldsymbol{u}_h, \mathcal{C}\boldsymbol{u}_h\right\rangle + \left\langle \boldsymbol{u}_h, \mathcal{M}^T\boldsymbol{p}_h\right\rangle + \mu\left\langle \boldsymbol{u}_h, \mathcal{D}\boldsymbol{u}_h\right\rangle + \rho\left\langle \boldsymbol{u}_h, \boldsymbol{F}_h\right\rangle \\ &= -\rho\left\langle \boldsymbol{u}_h, \mathcal{C}\boldsymbol{u}_h\right\rangle + \left\langle \mathcal{M}\boldsymbol{u}_h, \boldsymbol{p}_h\right\rangle + \mu\left\langle \boldsymbol{u}_h, \mathcal{D}\boldsymbol{u}_h\right\rangle + \rho\left\langle \boldsymbol{u}_h, \boldsymbol{F}_h\right\rangle.\end{aligned}$$

Since $\boldsymbol{u}_h$ is real and the convective terms have been discretised such that the discrete convective operator $\mathcal{C}$ is a skew-symmetric matrix, the first term on the right-hand side of this equation vanishes. Also, the second term is equal to zero because discrete mass is conserved. Hence, in the absence of an external force, the evolution of discrete kinetic energy is given by

$$\frac{dE_h}{dt} = \mu\left\langle \boldsymbol{u}_h, \mathcal{D}\boldsymbol{u}_h\right\rangle.$$

Since the discrete diffusive operator $\mathcal{D}$ is a negative-definite matrix, it follows that $dE_h/dt \leq 0$; energy is dissipated because of viscosity. Thus, in the absence of an external force, the spatial discretisation is guaranteed stable.

Note that stability could be proven since the symmetry properties of the discrete difference operators are inherited from the continuous differential operators. If, for example, only the discrete divergence operator contains apertures and the gradient operator is discretised with standard finite differences [1, 73], then the equality $\mathcal{P} = -\mathcal{M}^T$ is not longer valid and precautions might be necessary for a stable spatial discretisation.

2.3.4 Temporal Discretisation

In this section the continuity equation (2.19) and the momentum equations (2.20) are discretised in time. For the time discretisation the explicit forward Euler method is exploited. Using a superscript indicating the time level, this results in

$$\begin{aligned}\mathcal{M}\boldsymbol{u}_h^{n+1} &= 0, \\ \mathcal{V}\frac{\boldsymbol{u}_h^{n+1}-\boldsymbol{u}_h^n}{\delta t} + \mathcal{C}\left(\boldsymbol{u}_h^n\right)\boldsymbol{u}_h^n &= -\frac{1}{\rho}\left(-\mathcal{M}^T\boldsymbol{p}_h^{n+1} - \mu\mathcal{D}\boldsymbol{u}_h^n\right) + \boldsymbol{F}_h^n.\end{aligned} \tag{2.21}$$

Rearranging terms in the momentum equations gives

$$\boldsymbol{u}_h^{n+1} = \tilde{\boldsymbol{u}}_h^n + \delta t \mathcal{V}^{-1} \frac{1}{\rho} \mathcal{M}^T \boldsymbol{p}_h^{n+1}, \tag{2.22a}$$

where

$$\tilde{\boldsymbol{u}}_h^n = \boldsymbol{u}_h^n - \delta t \mathcal{V}^{-1} \left(\mathcal{C}\left(\boldsymbol{u}_h^n\right) \boldsymbol{u}_h^n - \frac{\mu}{\rho} \mathcal{D} \boldsymbol{u}_h^n - \boldsymbol{F}_h^n \right). \tag{2.22b}$$

The continuity equation is discretised at the new time level $n+1$ to ensure a divergence-free velocity field at the latter time level. Except for the pressure, all the terms in the momentum equations are discretised at the old time level n. Note that if the pressure gradient is discretised at the old time level as well, then both the continuity equation and the momentum equations predict a velocity field at the new time level which need not match. By discretising the pressure at the new time level, a correction to the intermediate vector field $\tilde{\boldsymbol{u}}_h^n$ is applied, such that the new velocity field is divergence free. The method to solve equations (2.21) and (2.22) is discussed in the next section.

2.3.5 Solution Method

First, the solution method for the discretised continuity equation and momentum equations is explained in the case that the entire flow domain is occupied by fluid. Then, some modifications are presented to account for the presence of a free surface. Hereto, some notation is introduced for sets of cells and cell faces with the same labels (for an explanation of the labeling see section 2.2.2). The set of all F cells is denoted by Ω_F, the set of all B cells by Ω_B, and so forth. Similarly, the set of all FF cell faces is denoted by Ω_{FF}, the set of all FB cell faces by Ω_{FB}, and so forth.

Without Free Surface

Assume that the flow domain is completely filled with liquid. In this case, every computational cell belongs to $\Omega_F \cup \Omega_B$ and every cell face to $\Omega_{FF} \cup \Omega_{FB} \cup \Omega_{BB}$. Further, assume that an initial velocity field $\boldsymbol{u}_h^n$ is given on $\Omega_{FF} \cup \Omega_{FB}$ (velocities at BB cell faces are not needed in the model — although, in the finite-volume method, a velocity does not have a fixed position at a cell face, but rather has a constant value throughout the control volume corresponding to that cell face, it is more convenient to speak of velocities located at cell faces). Since every F cell contains fluid and fluid is contained in F cells only, the discrete continuity equation (2.21) is solved on Ω_F. Every cell face having an edge aperture not equal to zero is an FF cell face. Hence, the discrete momentum equations (2.22) are solved on Ω_{FF}. This is done by first constructing a temporary vector field $\tilde{\boldsymbol{u}}_h^n$ on Ω_{FF} using equation (2.22b). Next, equation (2.22a) is substituted in (2.21), which gives a Poisson equation for the pressure on Ω_F

$$\mathcal{M}\mathcal{M}^T \boldsymbol{p}_h^{n+1} = -\frac{1}{\delta t} \rho \mathcal{V} \mathcal{M} \tilde{\boldsymbol{u}}_h^n. \tag{2.23}$$

This equation is solved iteratively with an SOR method (Gauss-Seidel with overrelaxation), where the relaxation parameter is automatically adjusted for optimal convergence behaviour [4]. A checkerboard ordering of the pressure is used to facilitate parallelisation

on a parallel computer platform. For the problems that are considered in this thesis, the SOR method performs adequately; a more modern solver does not increase the performance significantly [80]. Once the pressure at the new time level is known, the vector field $\tilde{\boldsymbol{u}}_h^n$ is corrected with the pressure gradient (equation (2.22a)) to obtain the velocity field on Ω_{FF} at the new time level. Note that velocities on Ω_{FB} do appear in the discretised continuity equation and in the discretised convective and diffusive terms (thus are needed in the construction of $\tilde{\boldsymbol{u}}_h^n$). In the continuity equation and convective terms, these velocities are always multiplied with a zero edge aperture, whence the actual value is not relevant. However, for the diffusive terms, velocities on Ω_{FB} have to be zero. Thus, these velocities are initialised to zero once and keep this value throughout the simulation.

With Free Surface

If only part of the flow domain is filled with liquid, then a computational cell and cell face belong to $\Omega_F \cup \Omega_B \cup \Omega_S \cup \Omega_E$ and $\Omega_{FF} \cup \Omega_{FB} \cup \Omega_{BB} \cup \Omega_{FS} \cup \Omega_{SB} \cup \Omega_{SS} \cup \Omega_{EB} \cup \Omega_{SE} \cup \Omega_{EE}$ respectively. In this case the presence of fluid is not restricted to F cells; also S cells contain fluid. On Ω_F, like above, the continuity equation is discretised and a Poisson equation for the pressure is formed. On Ω_S, boundary condition (2.6) for the pressure at the free surface is discretised (see section 2.3.6). These discretised equations (one in every S cell) are added to the system (2.23), giving an equation for the pressure on $\Omega_F \cup \Omega_S$. Because the pressure is solved in F and S cells, the momentum equations can be discretised on $\Omega_{FF} \cup \Omega_{FS} \cup \Omega_{SS}$. Velocities on $\Omega_{FB} \cup \Omega_{SB} \cup \Omega_{EB}$ are initialised to zero once since these are needed (or will be needed in a later time step) in the computation of the diffusive terms. Velocities on Ω_{BB} are not needed in the model. Near the free surface, velocities on $\Omega_{SE} \cup \Omega_{EE}$ may be needed in the discretised continuity equation and momentum equations. The computation of these velocities is discussed in section 2.3.6.

2.3.6 Free-Surface Boundary Conditions

In this section the discretisation of the boundary conditions for computing velocities on $\Omega_{SE} \cup \Omega_{EE}$ and the pressure in S cells is discussed.

Velocities on Ω_{SE}

When velocities at cell faces between an S and an E cell (SE velocities for short) appear in the discretised momentum equations, these velocities need a value. Also, some velocities on Ω_{SE} are required in the computation of velocities on Ω_{EE} (EE velocities for short) as is demonstrated below. By continuing the velocity field to S cells, all the velocities on Ω_{SE} are given a value. In other words, these velocities are computed by demanding conservation of mass (equation (2.8)) in S cells. However, contrary to F cells, the pressure Poisson equation is not formed in S cells. Note that if mass is not conserved in S cells, then diagonal entries of the coefficient matrix for convection are not equal to zero in rows corresponding to FS velocities, whence conservation of kinetic energy can not be guaranteed (see section 2.3.3).

If only one of the cell faces of an S cell belongs to Ω_{SE}, as is shown in the left of figure 2.13, then the other cell faces belong to $\Omega_{FS} \cup \Omega_{SS} \cup \Omega_{SB}$. All the velocities corresponding to these cell faces are either solved from the momentum equations or the

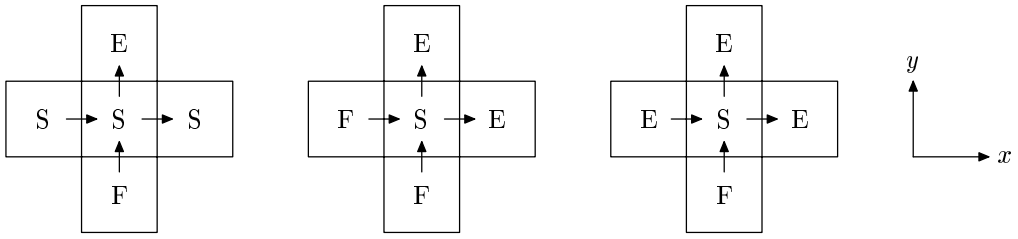

Figure 2.13: *Three possible label configurations around an S cell.*

no-slip boundary condition. Hence, the SE velocity can be solved from equation (2.8). If two or more cell faces of an S cell belong to Ω_{SE}, then conservation of mass is demanded in each Cartesian direction. Thus, for example, in the middle configuration in figure 2.13, the mass flux through each SE cell face is set equal to the mass flux through the opposite FS cell face. This is not possible if two SE cell faces face each other, such as in the right of figure 2.13. In this case, the two SE velocities are set to zero.

In three dimensions an extra complication arises. Consider therefore the middle and right-hand configurations in figure 2.13 with F cells in front of and behind (thus in z-direction) the central S cell. Now, it is not correct to prescribe conservation of mass in the x- and y-direction to solve the SE velocities. Indeed, the two FS velocities in z-direction, which are solved from the momentum equations, need not satisfy $\partial w/\partial z = 0$. Therefore, in these cases an amount $\partial w/\partial z$ is redistributed over the SE velocities such that mass is conserved in all S cells.

Velocities on Ω_{EE}

Velocities on Ω_{EE} are needed in the discrete momentum equations for neighbouring SS velocities (see figure 2.14). Since EE velocities are approximately tangential to the free

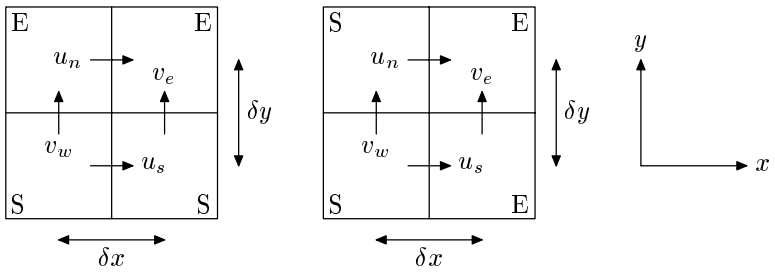

Figure 2.14: *Two possible label configurations in two dimensions near an EE velocity.*

surface, the tangential boundary condition (2.5) for the velocity is discretised for the computation of these velocities. In this boundary condition derivatives of the velocity normal and tangential to the free surface in the direction tangential and normal to the free surface appear. For an arbitrary orientation of the free surface these derivatives are difficult to compute on a Cartesian grid. Hence, equation (2.5) is split in Cartesian directions. Thus, in two dimensions, every EE velocity that is needed in the model is

computed by discretising

$$\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) = 0. \tag{2.24}$$

Equation (2.24) is discretised using standard finite differences. For example, for the left-hand configuration in figure 2.14, the EE velocity u_n is solved from

$$\mu \left(\frac{u_n - u_s}{\delta y} + \frac{v_e - v_w}{\delta x} \right) = 0.$$

In this equation SE velocities v_e and v_w are needed. Hence, it is important that SE velocities are computed before EE velocities. The SS velocity u_s is solved from the discrete momentum equations. In extraordinary occasions it is possible that, in two dimensions, an EE velocity is surrounded by two SS velocities. In this case, equation (2.24) is discretised twice and the resulting values are averaged (a weighted average is taken on a stretched grid) to produce a final value for the EE velocity.

In three dimensions the computation of EE velocities is more complicated since the number of SS velocities around an EE velocity may vary from zero up to four. Further, the tangential boundary condition (2.5) separates in three (Cartesian) equations, namely

$$\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) = 0, \quad \mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) = 0, \quad \text{and} \quad \mu \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) = 0. \tag{2.25}$$

Of course, if an EE velocity has no neighbouring SS velocities, then this EE velocity is not needed in the model. If it has one neighbouring SS velocity, then one of the equations in (2.25), dependent on the direction of the EE velocity and the SE velocities, is discretised. For example, for configurations (a) and (b) in figure 2.15, the third respectively first equation in (2.25) is discretised for computing the EE velocity. If an EE velocity has two neighbouring SS velocities, then either the two SS velocities are located in different Cartesian directions (configuration (c) in figure 2.15) or the two SS velocities face each other (configuration (d)). In both cases, for each of the SS velocities a value for the EE

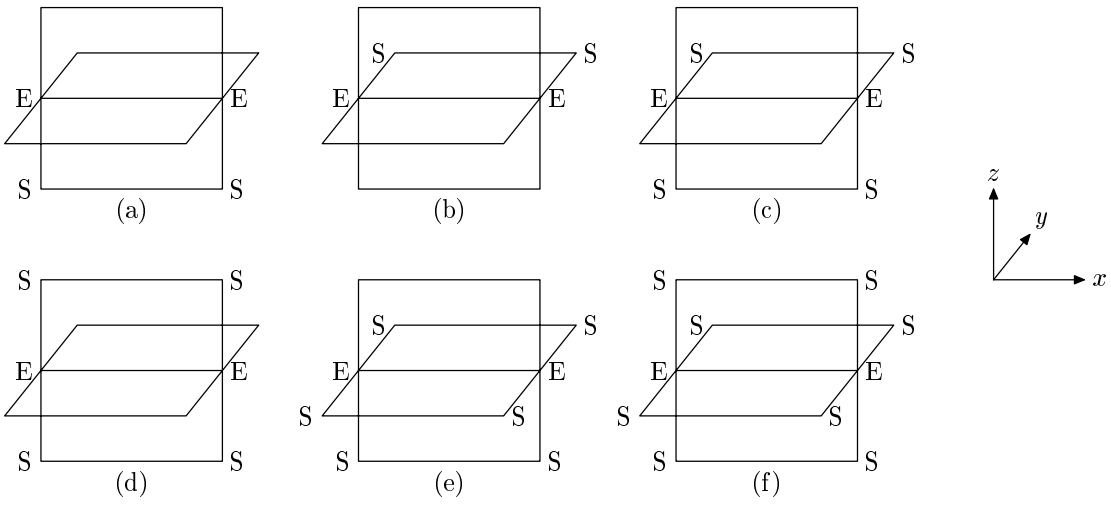

Figure 2.15: *Six possible label configurations near an EE velocity in three dimensions.*

velocity is computed by discretising one of the equations in (2.25). Then, based on these two values, a final value for the EE velocity is computed using weighted averaging. In the case of three neighbouring SS velocities the same procedure as with one SS velocity is applied; the two SS velocities facing each other are, because of symmetry reasons, not used in the computation of the EE velocity. Indeed, in configuration (e) in figure 2.15, the free surface is approximately parallel to the xy-plane, whence only the third equation in (2.25) is discretised. Finally, if an EE velocity is surrounded by four SS velocities (configuration (f)), then for each SS velocity an equation in (2.25) is discretised. The four values for the EE velocity are averaged to produce a final value for the EE velocity.

Pressure in S Cells

In S cells a value for the pressure is needed, which is provided by boundary condition (2.6) for the pressure at the free surface. Since the second term ($2\ \mu \partial u_n / \partial \boldsymbol{n}$) on the left-hand side in this equation is usually small compared to the other terms (especially in a micro-gravity environment, where capillary forces dominate, but also because of the small value of the viscosity that is used in this thesis [36]), this term is neglected in the numerical model, which leaves

$$p = p_0 - \sigma\kappa. \tag{2.26}$$

Here, p is the pressure at the free surface, p_0 is the pressure in the ambient air, σ is the surface tension of the fluid, and κ denotes the mean curvature of the free surface. Since this equation is discretised in S cells, the mean curvature of the free surface needs to be computed in every S cell (this procedure is explained below). Once the curvature is known, the pressure at the free surface is easily found using equation (2.26). In the finite-volume method it is assumed that the pressure has a constant value throughout each computational cell. Hence, it is possible to define the pressure in an S cell to be the pressure at the free surface in this cell. Thus, equation (2.26) is added unchanged to the pressure Poisson equation. Another option is to define the pressure in an S cell based on interpolation of the pressure at the free surface and the pressure in a neighbouring F cell [35, 49]. This is especially beneficiary for simulation of liquid sloshing in a micro-gravity environment, where the shape of the free surface drives the flow. This method is illustrated in figure 2.16. Based on the orientation of the free surface (in two dimensions horizontal or vertical; see the next paragraph for more details), an F cell is determined, which is used for computing the pressure in the centre of the S cell. If the distance between

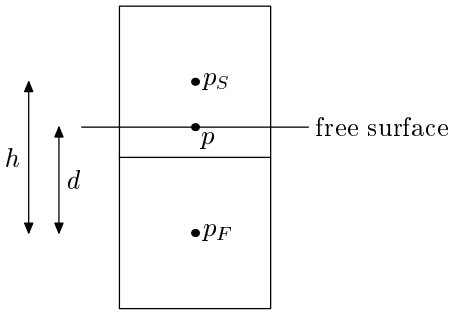

Figure 2.16: *Interpolation of the free-surface pressure to the centre of an S cell.*

the centre of the F cell and the free surface is equal to d and the distance between the centres of the F and S cell is h, then the pressure p_S in the centre of the S cell is given by

$$p_S = \eta p + (1-\eta)\, p_F = \eta\,(p_0 - \sigma\kappa) + (1-\eta)\, p_F,$$

where p_F is the pressure in the (centre of the) F cell and $\eta = h/d$.

Curvature of the Free Surface

For applying the boundary condition for the pressure at the free surface, in every S cell the mean curvature κ of the free surface has to be computed. If the free surface is described by a level-set function $S(x,y,t) = 0$, then the mean curvature of the free surface is given by $\kappa = \nabla \cdot \boldsymbol{n}$, where $\boldsymbol{n} = \nabla S/|\nabla S|$ is the normal at the free surface. In this thesis the free surface is described locally in every S cell using a height function. In two dimensions (the situation in three dimensions is explained below), based on the orientation of the free surface in the S cell, either a horizontal or vertical height function is defined using the VOF fractions in a 3×3 block of cells around the S cell. This is illustrated in figure 2.17. In this figure the free surface is approximately vertical, which

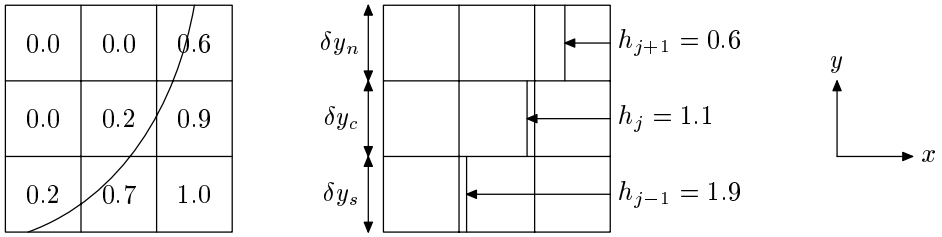

Figure 2.17: *Free surface and VOF fractions (left) and discrete, vertical local height function corresponding to the centre S cell (right).*

is concluded by inspecting differences of VOF fractions (in absolute value). In this case the difference of the eastern and western VOF fraction equals $|0.9 - 0.0| = 0.9$, which is larger than $|0.0 - 0.7| = 0.7$, the difference of the northern and southern VOF fraction. Another method for determining the approximate orientation of the free surface is based on the reconstructed normal $\boldsymbol{n} = (n_x, n_y)^T$ of the free surface (see also section 2.4.1). If $|n_x \delta x| > |n_y \delta y|$ (where δx and δy denote the mesh size of the S cell in x- and y-direction respectively), a vertical height function is defined. Thus, for the centre S cell in figure 2.17, a vertical local height function $x = h(y,t)$ is defined. In terms of the level-set function this corresponds to $S(x,y,t) \equiv h(y,t) - x = 0$. In this case the mean curvature can be written as

$$\kappa = \frac{\partial}{\partial y}\left(\frac{\partial h/\partial y}{\sqrt{1 + (\partial h/\partial y)^2}}\right). \tag{2.27}$$

The local height function h is discretised by adding the VOF fractions in three rows (see figure 2.17). Next, at grid lines $y = j - \frac{1}{2}$ and $y = j + \frac{1}{2}$, the derivative $\partial h/\partial y$ is discretised using standard finite differences, *i.e.*

$$\left.\frac{\partial h}{\partial y}\right|_{j-\frac{1}{2}} \doteq \frac{h_j - h_{j-1}}{\frac{1}{2}\delta y_s + \frac{1}{2}\delta y_c} \equiv h'_s \quad \text{and} \quad \left.\frac{\partial h}{\partial y}\right|_{j+\frac{1}{2}} \doteq \frac{h_{j+1} - h_j}{\frac{1}{2}\delta y_c + \frac{1}{2}\delta y_n} \equiv h'_n.$$

With these discretisations the term between brackets in equation (2.27) can be computed at the same positions $y = j \pm \frac{1}{2}$, whence the discrete mean curvature in the centre S cell is given by

$$\kappa \doteq \frac{1}{\delta y_c} \left(\frac{h'_n}{\sqrt{1 + {h'_n}^2}} - \frac{h'_s}{\sqrt{1 + {h'_s}^2}} \right).$$

Although computing the curvature is slightly more complicated in three dimensions, the procedure is the same as in two dimensions. The normal at the free surface in an S cell is classified as directed primarily in either the x-, y-, or z-direction. Based on this orientation, a local height function is formed, *e.g.* $S(x, y, z, t) \equiv h(x, y, t) - z = 0$. In this case, the mean curvature equals

$$\kappa = \frac{\partial}{\partial x} \left(\frac{\partial h/\partial x}{\sqrt{1 + (\partial h/\partial x)^2 + (\partial h/\partial y)^2}} \right) + \frac{\partial}{\partial y} \left(\frac{\partial h/\partial y}{\sqrt{1 + (\partial h/\partial x)^2 + (\partial h/\partial y)^2}} \right). \quad (2.28)$$

Since this expression contains second-order mixed derivatives, a local height is needed in 9 columns of cells around the centre S cell (see figure 2.18). For computing the discrete

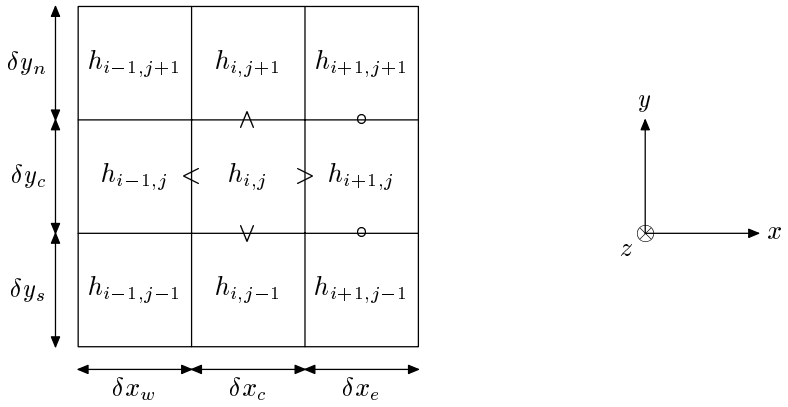

Figure 2.18: *Notation used in the formation of a three-dimensional local height function for the centre S cell.*

height function in these 9 columns, VOF fractions are added as in two dimensions (hereto the 26 cells surrounding the S cell and the S cell itself are used). The curvature κ in (2.28) is then discretised as

$$\kappa \doteq \frac{n_e - n_w}{\delta x_c} + \frac{n_n - n_s}{\delta y_c},$$

where n_e and n_w are discretisations of the first term between brackets in (2.28) at the positions indicated by $>$ and $<$ in figure 2.18 respectively and n_n and n_s are discretisations of the second term between brackets at the positions indicated by $\wedge$ and $\vee$ respectively. For computing n_e, for example, both $\partial h/\partial x$ and $\partial h/\partial y$ are needed at $>$. The former is discretised as

$$\left.\frac{\partial h}{\partial x}\right|_{>} \doteq \frac{h_{i+1,j} - h_{i,j}}{\frac{1}{2}\delta x_c + \frac{1}{2}\delta x_e}.$$

For the latter, $\partial h/\partial y$ is discretised at four horizontal cell faces (indicated by $\wedge$, $\vee$, and $\circ$) using expressions similar to the one for $\partial h/\partial x$. These four values are then averaged to obtain a value for $\partial h/\partial y$ at $>$.

Contact Angle

At the intersection of the free surface and the solid body, which is called the contact line, a boundary condition is needed for computing the mean curvature at the free surface. This is given by a static contact angle θ, which is the angle between the normal of the free surface and the normal of the solid body at the contact line. The discretisation of the contact angle is first explained in two dimensions, thereafter in three dimensions.

In two dimensions the contact line is a set of points. A typical configuration is sketched in the left of figure 2.19, where the contact line is indicated by $\bullet$ and the normal of the solid body (pointing into the flow domain) is denoted by $\boldsymbol{n}_b$ (see also [44], where only staircase approximations of the solid body are used for prescribing the contact angle). Now, for a given contact angle, two possible directions of the normal $\boldsymbol{n}_s$ of the free surface

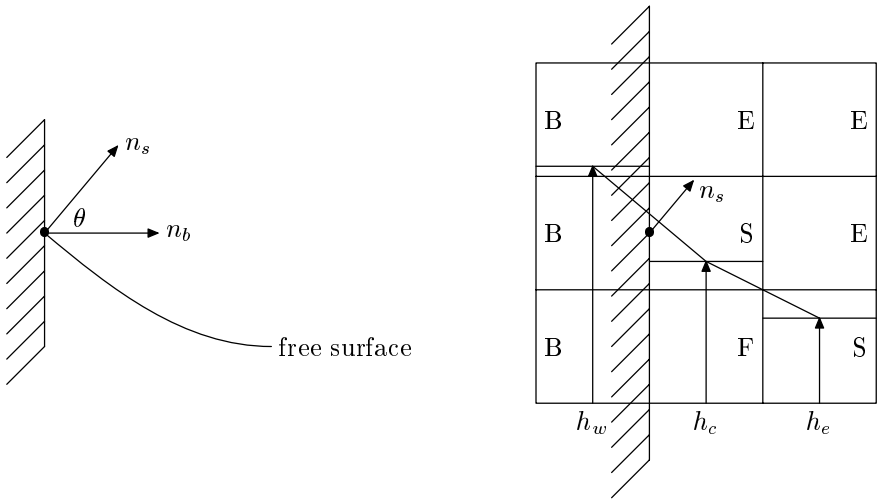

Figure 2.19: *Left: The contact angle θ is the angle between the normal $\boldsymbol{n}_b$ of the solid boundary and the normal $\boldsymbol{n}_s$ of the free surface at the contact line (which is indicated by $\bullet$). Right: Horizontal local height function near a vertical wall.*

remain. Based on the location of the liquid a definite choice of the normal $\boldsymbol{n}_s$ is made. For example, in figure 2.19 the liquid is positioned in the lower part, whence the normal of the free surface points upward. With this normal vector, the local height function in S cells near the solid body can be computed as is demonstrated in the right of figure 2.19. In this figure the free surface is approximately horizontal, whence a horizontal height function is formed for the centre S cell. Since the cell on the left of this cell is a B cell, no liquid height can be computed in the left-hand column of cells. Hence, in this column, the height is set to a value h_w, such that the line through h_w and h_c is perpendicular to the normal vector $\boldsymbol{n}_s$. Note that in this example the boundary of the solid body is vertical and the free surface is approximately horizontal. In such a case, where the normal of the solid boundary and the normal of the free surface are classified as being perpendicular, it

is sufficient to consider SB cell faces for applying the contact angle in computing the local height function. If the normal of the solid boundary and the normal of the free surface at the contact line have approximately the same direction, the occurrence of an SB cell face is not sufficient for applying the contact angle. This is illustrated in figure 2.20, where a vertical height function for the centre S cell is computed near a vertical wall. In this

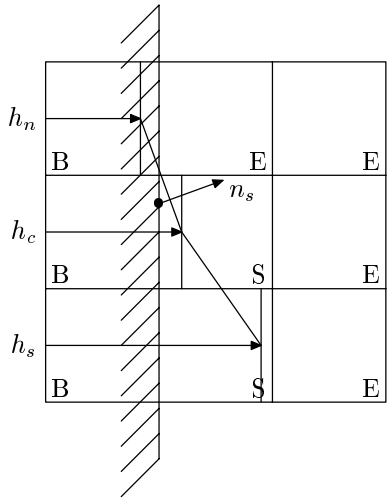

Figure 2.20: *Vertical local height function near a vertical wall.*

example the presence of an E cell above the S cell is necessary in order to discretise the contact angle. This is done by setting the height in the top row to a value h_n, such that the line connecting h_n and h_s is perpendicular to $\boldsymbol{n}_s$.

Some care has to be taken in computing the height function near the boundary of the solid body. For example, in the bottom row in figure 2.20, it is not sufficient to add the VOF fractions in the three cells since then only the contribution of the S cell is accounted for. Hence, instead of adding the VOF fractions, the quantity $F^s + 1 - F^b$ (which is 1 in B cells that are completely filled with liquid) is added in the three cells for computing the discrete height.

Determining the normal $\boldsymbol{n}_s$ in three dimensions is more complicated than in two dimensions (see figure 2.21). Indeed, if the normal $\boldsymbol{n}_b$ at the solid boundary is given, then an infinite number of vectors make an angle θ with this normal vector. Two of these vectors are perpendicular to the tangent $\boldsymbol{t}_s$ of the contact line and therefore are candidates for $\boldsymbol{n}_s$. Based on the location of the liquid (like in two dimensions), a final decision for the normal of the free surface can be made. Thus, to be able to determine the direction of the normal of the free surface, both the normal of the solid boundary and the tangent to the contact line are needed. For the computation of the former a technique described in section 2.4.1 (for the computation of normals at the free surface) is used. The computation of the tangent $\boldsymbol{t}_s$ is based on the local height function. Hereto, first, an approximation of the direction of the contact line is determined by looking at the approximate Cartesian directions of the solid boundary and the free surface. For example, if the largest component of the normal of the solid boundary is the x-component and if the local height function for the S cell under consideration is formed in z-direction, then a first guess for $\boldsymbol{t}_s$ is given by $\left(0, 1, h'_y\right)$, where h'_y is a finite-difference approximation

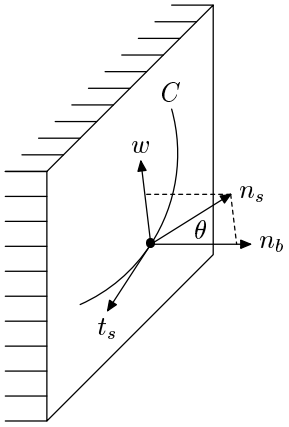

Figure 2.21: *In three dimensions the normal $\boldsymbol{n}_s$ of the free surface makes an angle θ with the normal $\boldsymbol{n}_b$ of the solid boundary and lies in the plane spanned by $\boldsymbol{n}_b$ and $\boldsymbol{w} = \boldsymbol{t}_s \times \boldsymbol{n}_b$, where $\boldsymbol{t}_s$ is the tangent to the contact line C.*

of $\partial h/\partial y$. Next, $\boldsymbol{t}_s$ is found by projecting this initial guess on the plane with normal $\boldsymbol{n}_b$. Finally, the normal $\boldsymbol{n}_s$ of the free surface at the contact line is given by a linear combination of $\boldsymbol{n}_b$ and $\boldsymbol{t}_s \times \boldsymbol{n}_b$, such that $\boldsymbol{n}_s$ makes an angle θ with $\boldsymbol{n}_b$ and points from the liquid into the air.

Similar to the technique described in two dimensions, the normal of the free surface at the contact line is used for constructing fictitious heights in B cells. Hereto, two different cases are considered. In the first case, the approximate direction of the solid boundary (the largest component of $\boldsymbol{n}_b$) is not equal to the approximate direction of the free surface (in two dimensions this situation corresponds to figure 2.19). In this case, if an S cell for which the curvature needs to be computed has a neighbouring B cell, then a vector $\boldsymbol{n}_s$ is computed in the S cell following the procedure described above. This vector is used for computing a local height in the B cell. Note that in three dimensions an S cell may be surrounded by several B cells. In such a situation, a fictitious height is computed in every B cell using the same vector $\boldsymbol{n}_s$. For example, in figure 2.22, if a height function is constructed in z-direction, then in the northern, western, and north-western B cell

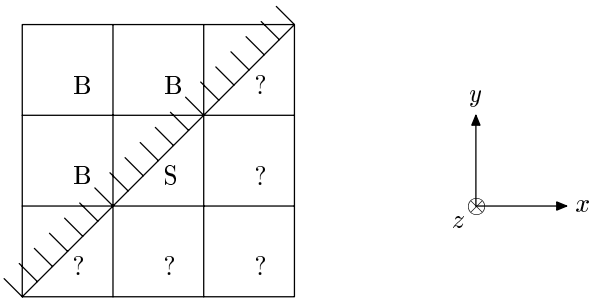

Figure 2.22: *Top view of label configuration around an S cell. The local height function in the centre S cell is constructed in z-direction, whence fictitious heights are required in the B cells.*

a fictitious height is computed using the normal of the free surface at the contact line. In the cells indicated with a question mark (either F, S, or E cells) a regular height is computed using the volume-of-fluid function F^s. In the second case, the approximate direction of the solid boundary is equal to the approximate direction of the free surface. In this case the situation is in fact two-dimensional and figure 2.20 is applicable.

2.4 Free-Surface Reconstruction and Advection

At the end of a time cycle, when a velocity field has been computed throughout the computational grid, the free surface needs to be advected. If the location of the free surface at time t is given by

$$S(x, y, z, t) = 0, \tag{2.29}$$

then its temporal evolution follows from

$$\frac{DS}{Dt} \equiv \frac{\partial S}{\partial t} + (\boldsymbol{u} \cdot \nabla) S = 0, \tag{2.30}$$

stating that the free surface propagates with the liquid velocity. In this thesis the location of the free surface is not described explicitly by an equation (2.29). Instead, a free-surface aperture or volume-of-fluid (VOF) function F^s is defined, denoting the fractional volume of a computational cell that is occupied by fluid (see also section 2.2.1). Based on the VOF function, the free surface is reconstructed, which is explained in section 2.4.1. Then, the reconstructed free surface is advected in time, giving a VOF function at the new time level (this is explained in section 2.4.2).

2.4.1 Piecewise-Linear Interface Reconstruction

In this section a method for reconstructing the free surface is discussed. In every cell with a free-surface aperture F^s between zero and F^b (such a cell will be referred to as a mixed cell) a linear approximation, say

$$n_x x + n_y y + n_z z = c, \tag{2.31}$$

of the free surface is constructed. For this, first, the normal $\boldsymbol{n} = (n_x, n_y, n_z)^T$ of the free surface is approximated. Next, the plane constant c is computed such that the fractional volume of the computational cell for which $n_x x + n_y y + n_z z > c$ is equal to F^s. First, the method is explained for cells not close to the solid boundary. Then, some modifications for cells near the solid boundary are presented.

Computation of the Normal

For computing an approximation of the normal of the free surface in a mixed cell, consider the VOF fractions in the mixed cell itself and its 8 nearest neighbours (in three dimensions its 26 nearest neighbours); see figure 2.23 for an illustration. Analytically, the normal of the free surface is given by $\boldsymbol{n} = \nabla F^s / |\nabla F^s|$. An approximation of the interface normal

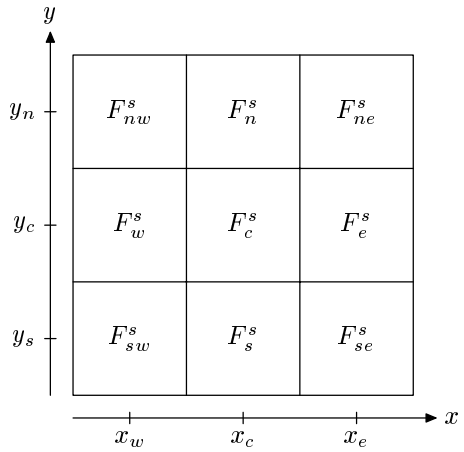

Figure 2.23: *Notation used in computing the normal of the free surface in the centre cell.*

is found by discretising this equation with respect to all the neighbours of the mixed cell, *i.e.*

$$\begin{pmatrix} x_e - x_c & 0 \\ x_e - x_c & y_n - y_c \\ 0 & y_n - y_c \\ x_w - x_c & y_n - y_c \\ x_w - x_c & 0 \\ x_w - x_c & y_s - y_c \\ 0 & y_s - y_c \\ x_e - x_c & y_s - y_c \end{pmatrix} \begin{pmatrix} \tilde{n}_x \\ \tilde{n}_y \end{pmatrix} = \begin{pmatrix} F^s_e - F^s_c \\ F^s_{ne} - F^s_c \\ F^s_n - F^s_c \\ F^s_{nw} - F^s_c \\ F^s_w - F^s_c \\ F^s_{sw} - F^s_c \\ F^s_s - F^s_c \\ F^s_{se} - F^s_c \end{pmatrix}. \tag{2.32}$$

In this equation the normal is not normalised yet (which is indicated by the tilde symbol). This system of linear equations is written in matrix-vector form as $\mathcal{A}\tilde{\boldsymbol{n}} = \boldsymbol{b}$. By multiplying both the left-hand side and the right-hand side with $\mathcal{A}^T$, a 2×2 (3×3 in three dimensions) linear system for $\tilde{\boldsymbol{n}}$ results, which is solved using Gaussian elimination. The normal of the free surface is then computed as $\boldsymbol{n} = \tilde{\boldsymbol{n}}/|\tilde{\boldsymbol{n}}|$.

Computation of the Plane Constant

Given the normal $\boldsymbol{n}$ of the free surface in a mixed cell, the plane constant c in equation (2.31) is computed such that the fractional volume of the mixed cell for which $n_x x + n_y y + n_z z > c$ equals the VOF value F^s in that cell. The correct value of c is found using a bisection method. For computing initial lower and upper bounds in this iterative method, first, in all vertices (four in two dimensions and eight in three dimensions) of the mixed cell the value of c and the corresponding VOF value is determined (the method for computing the VOF value for a given normal and plane constant is explained below). See figure 2.24 for a graphical explanation in two dimensions. For example, the line with normal $(n_x, n_y)^T$ passing through the north-eastern vertex is given by $n_x x + n_y y = c_{ne}$, where $c_{ne} = n_x x_e + n_y y_n$, and the corresponding VOF value F_{ne} is approximately equal to 0.75 in this figure (if the fluid is located below the line). By comparing the VOF values corresponding to the vertices of the mixed cell to the actual VOF value F^s, a lower bound

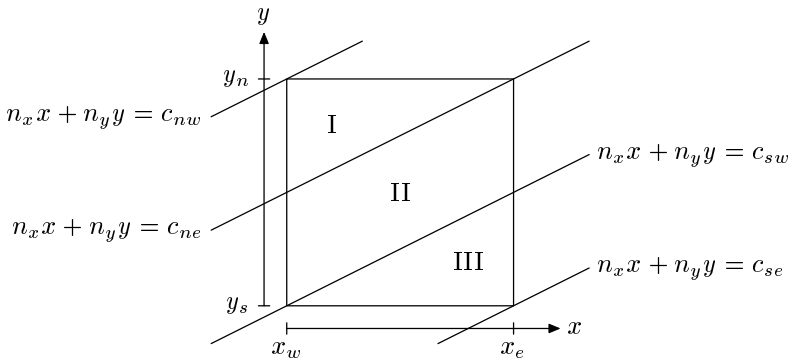

Figure 2.24: *For computation of the plane constant a lower and upper bound is computed by comparing the VOF values corresponding to the lines passing through the vertices of a mixed cell to the actual value F^s.*

c_l and an upper bound c_u for the value of the plane constant c can be found. Then, for the intermediate value $c_m = (c_l + c_u)/2$ the corresponding VOF value is computed and compared to F^s. Based on this comparison, either c_l or c_r is set to c_m. This process is repeated until the difference between the computed VOF value and F^s is smaller than a predefined tolerance.

Of course, an iterative method for finding the plane constant is not necessary in two dimensions. Indeed, the correct value of c can be found by solving a linear equation (in the parallelogram region II in figure 2.24) or a quadratic equation (in the triangular regions I and III). However, in three dimensions, five different regions can be distinguished and in two of these regions a cubic equation has to be solved for computing the VOF value. Since in these two regions an iterative method is needed, this iterative method is applied in all the regions for reasons of consistency. An obvious drawback of this approach is the extra amount of computing time that is required for the iterative method [67]. However, this iteration is performed in mixed cells only, which generally form a two-dimensional subset in three-dimensional space. Hence, the computing time required for this iteration is negligible compared to the iteration method for solving the pressure Poisson equation (see section 2.3.5). The same reason justifies the choice of the bisection method rather than a more efficient iterative method. Moreover, the bisection method is guaranteed to converge.

Computation of the VOF Value for Given Normal and Plane Constant

In the iterative method described above, the fractional volume of a mixed cell that is truncated by an equation (2.31) has to be computed for a given normal $\boldsymbol{n}$ of the free surface and a given plane constant c. This is explained in two dimensions using figure 2.25. The shaded region Ω represents the part of the cell for which $n_x x + n_y y > c$. The volume V of this region is given by (see also [19, 45])

$$V = \int_\Omega dV = \frac{1}{2} \int_\Omega \nabla \cdot \begin{pmatrix} x \\ y \end{pmatrix} dV = \frac{1}{2} \int_\Omega \nabla \cdot \begin{pmatrix} x - cn_x \\ y - cn_y \end{pmatrix} dV.$$

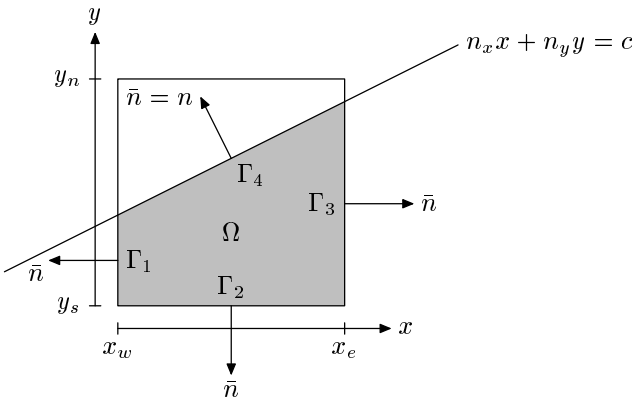

Figure 2.25: *For a given normal* $\boldsymbol{n}$ *and plane constant* c*, the volume of* Ω *(the shaded region) is computed using a boundary integral over* $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \cup \Gamma_4$*, which can be rewritten as a boundary integral over the cell faces.*

Using the divergence theorem, this last integral is transformed into a boundary integral over the boundary Γ of Ω, which results in

$$V = \frac{1}{2} \oint_{\Gamma} \begin{pmatrix} x - cn_x \\ y - cn_y \end{pmatrix} \cdot \bar{\boldsymbol{n}} \, dS,$$

where $\bar{\boldsymbol{n}}$ is the outward-pointing normal on Γ. The boundary Γ consists of four parts Γ_1, Γ_2, Γ_3, and Γ_4. On the former three, the normal $\bar{\boldsymbol{n}}$ is aligned with the x- or y-axis. On the latter, $\bar{\boldsymbol{n}}$ is equal to $\boldsymbol{n}$. Hence, the volume V is equal to

$$V = -\frac{x_w - cn_x}{2} \int_{\Gamma_1} dy - \frac{y_s - cn_y}{2} \int_{\Gamma_2} dx + \frac{x_e - cn_x}{2} \int_{\Gamma_3} dy + \frac{1}{2} \int_{\Gamma_4} (n_x x + n_y y - c) \, dS.$$

In this equation the last integral vanishes since $n_x x + n_y y = c$ on Γ_4. The remaining integrals are rewritten as integrals over the cell faces, *i.e.*

$$\begin{aligned} V = &- \frac{x_w - cn_x}{2} \int_{y_s}^{y_n} \mathbf{1}_{\{n_x x_w + n_y y > c\}} \, dy - \frac{y_s - cn_y}{2} \int_{x_w}^{x_e} \mathbf{1}_{\{n_x x + n_y y_s > c\}} \, dx \\ &+ \frac{x_e - cn_x}{2} \int_{y_s}^{y_n} \mathbf{1}_{\{n_x x_e + n_y y > c\}} \, dy + \frac{y_n - cn_y}{2} \int_{x_w}^{x_e} \mathbf{1}_{\{n_x x + n_y y_n > c\}} \, dx, \end{aligned} \tag{2.33}$$

where the last term, an integral over the northern cell face, is added to obtain a formula that can be applied in every mixed cell. Since $(x, y) \mapsto n_x x + n_y y$ is linear in x and y, the integrals that appear in equation (2.33) are easily computed. For example, the integral over the western cell face is computed as

$$\int_{y_s}^{y_n} \mathbf{1}_{\{n_x x_w + n_y y > c\}} \, dy = (y_n - y_s) \frac{\max\{c_{nw}, c_{sw}\}}{\max\{c_{nw}, c_{sw}\} - \min\{c_{nw}, c_{sw}\}},$$

where $c_{nw} = n_x x_w + n_y y_n$ and $c_{sw} = n_x x_w + n_y y_s$. In this equation it is assumed that $c_{nw} c_{sw} < 0$. If this is not the case, then either $c_{nw} > 0$ and $c_{sw} > 0$, whence the integral equals $y_n - y_s$, or $c_{nw} \leq 0$ and $c_{sw} \leq 0$, whence the integral equals zero. The advantage of this method is that the integral over the reconstructed interface is removed; only integrals over the cell faces have to be computed. Further, the method is generic in the sense that no different cases have to be considered; equation (2.33) is applied in every mixed cell.

In three dimensions a similar method is used. The volume V of a cell that is truncated by an equation (2.31) is written as a boundary integral over the six cell faces, *i.e.*

$$\begin{aligned} V = &- \frac{x_l - cn_x}{3} \int_{y_l}^{y_r} \int_{z_l}^{z_r} \mathbf{1}_{\{n_y y + n_z z > c - n_x x_l\}} \, dydz - \frac{y_l - cn_y}{3} \int_{x_l}^{x_r} \int_{z_l}^{z_r} \mathbf{1}_{\{n_x x + n_z z > c - n_y y_l\}} \, dxdz \\ &- \frac{z_l - cn_z}{3} \int_{x_l}^{x_r} \int_{y_l}^{y_r} \mathbf{1}_{\{n_x x + n_y y > c - n_z z_l\}} \, dxdy + \frac{x_r - cn_x}{3} \int_{y_l}^{y_r} \int_{z_l}^{z_r} \mathbf{1}_{\{n_y y + n_z z > c - n_x x_r\}} \, dydz \\ &+ \frac{y_r - cn_y}{3} \int_{x_l}^{x_r} \int_{z_l}^{z_r} \mathbf{1}_{\{n_x x + n_z z > c - n_y y_r\}} \, dxdz + \frac{z_r - cn_z}{3} \int_{x_l}^{x_r} \int_{y_l}^{y_r} \mathbf{1}_{\{n_x x + n_y y > c - n_z z_r\}} \, dxdy, \end{aligned}$$

where $[x_l, x_r] \times [y_l, y_r] \times [z_l, z_r]$ are the dimensions of the mixed cell under consideration. For each of the six integrals in this equation, the two-dimensional method that has been described above is applied, *i.e.* each double integral is replaced by an equation similar to (2.33). Note that the method in this section is mathematically more elegant (and on a parallel computer more efficient) than a case-by-case method for computing VOF fractions as is done in the original method of Youngs [96].

Near the Solid Boundary

The methods for computing the normal of the free surface and the VOF value for a given normal and plane constant require some modifications in the vicinity of the solid boundary.

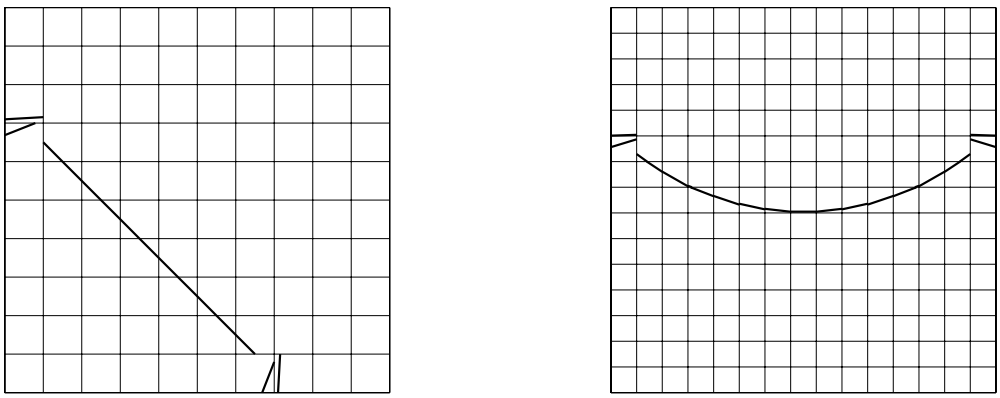

Figure 2.26: *Reconstruction of a line (left) and part of a circle (right). In B cells the value of F^s is set to zero and these cells are taken into account in system* (2.32).

The basic procedure for the computation of the normal is the same. However, near the solid boundary, some cells neighbouring the mixed cell under consideration may be entirely closed for flow ($F^b = 0$). A typical example is shown in figure 2.19, where the three cells in the left-hand column are B cells. In these cells the value of F^s is equal to zero and should not be used in the computation of the normal as is illustrated in figure 2.26. In this figure, a straight line (on the left) and part of a circle (on the right) are reconstructed on Cartesian grids with 10×10 and 15×15 cells respectively (the fluid is located below the line or circle). The value of F^s is equal to zero in B cells. Clearly, the reconstructed normal is not correct in mixed cells adjacent to B cells. If the contribution of B cells is removed from the system (2.32), then the reconstructed normals in mixed cells near the solid boundary are more accurate as is demonstrated in figure 2.27. Instead

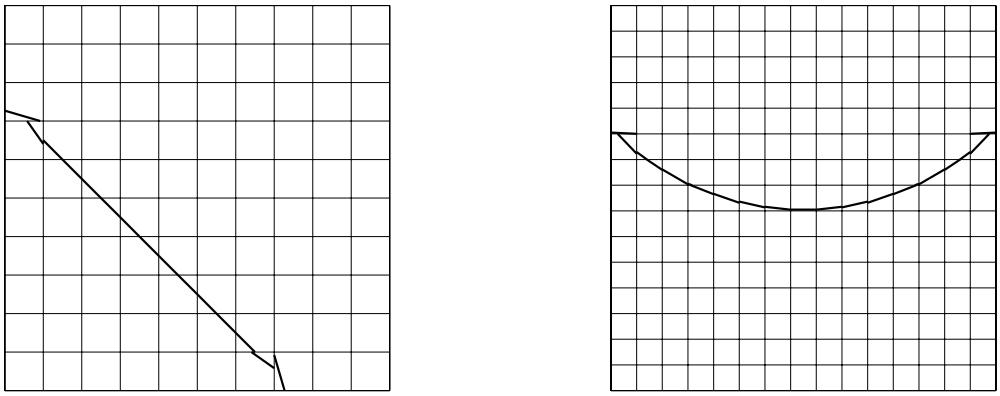

Figure 2.27: *Reconstruction of a line (left) and part of a circle (right). The contribution of B cells is removed from system* (2.32).

of ignoring B cells in computing the normal of the free surface, it is possible to take these B cells into account with suitable values of F^s. For example, if a straight line passes through the centre of the centre cell in figure 2.19 and is continued in the solid body, then, using the notation of figure 2.23, it follows that

$$F^s_{nw} + F^s_{se} = 1, \quad F^s_w + F^s_e = 1, \quad \text{and} \quad F^s_{sw} + F^s_{ne} = 1.$$

This observation motivates the following choice for assigning VOF values to B cells: if cell (i, j, k) is a mixed cell wherein a normal has to be reconstructed and cell (l, m, n) is a B cell (and one of the 26 neighbours of the mixed cell), then the VOF value $F^s_{l,m,n}$ in this B cell is set to $1 - F^s_{i+i-l,j+j-m,k+k-n}$. The results of this method are shown in figure 2.28. From figures 2.27 and 2.28 it becomes clear that in some mixed cells near the solid boundary removing B cells from system (2.32) gives more accurate normals and in other mixed cells "mirrored" F^s values result in a higher accuracy. Closer inspection revealed that the former method is best applied in cells with a large value of F^s and the latter in the case of small F^s values.

For computing the plane constant, the same iterative method that has been described above is used. No changes are required in the vicinity of the solid boundary. However, at every iteration level in this method, the fractional volume of a mixed cell that is truncated by the free surface has to be computed. This was done by changing a volume integral into a carefully chosen boundary integral, such that the integral along the free surface

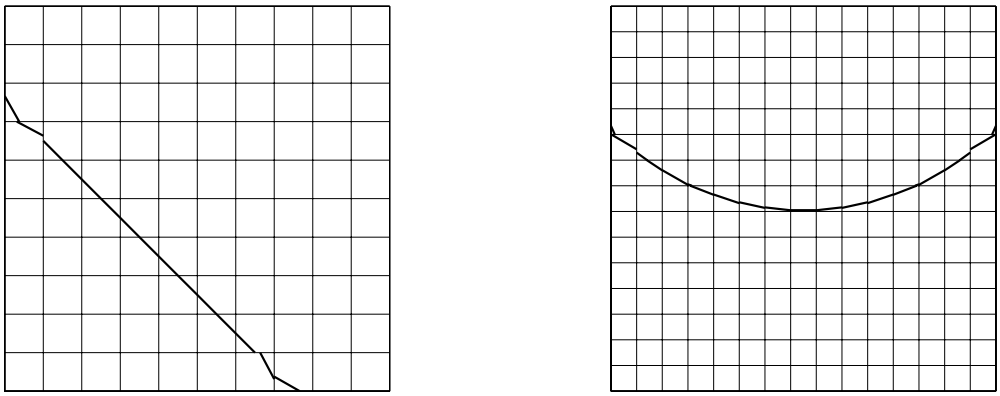

Figure 2.28: *Reconstruction of a line (left) and part of a circle (right). The contribution of B cells is taken into account, but "mirrored" values of F^s are assigned to these cells.*

vanished. If the mixed cell is cut by the solid body, this procedure is more difficult as is illustrated in figure 2.29 on the left. In this figure the dark-shaded region is not open for flow. The volume integral for computing the volume of Ω can be rewritten in such a way that the boundary integral along Γ_5 (the free surface) vanishes, but then the contribution of Γ_4 (the solid boundary) is not equal to zero. Thus, in this case, the area of Γ_4 needs to be computed, which is relatively difficult (compared to the computation of VOF values in mixed cells that are not cut by the solid body). Hence, in cut cells, the solid boundary is approximated such that it is aligned with one of the Cartesian directions as is shown in figure 2.29 on the right. The configuration in this figure is similar to the one in figure 2.25; only the value of x_e is changed.

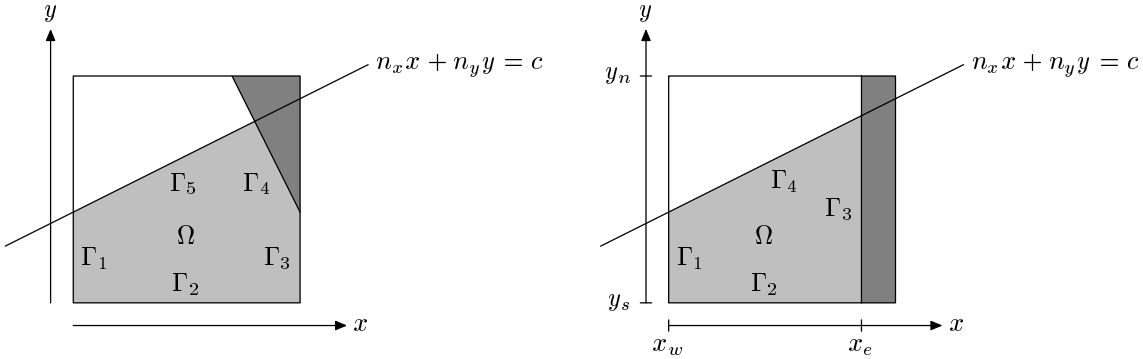

Figure 2.29: *Computation of the VOF value for a mixed cell that is cut by the solid boundary; the solid body is approximated by a line that is aligned with one of the Cartesian directions.*

2.4.2 Free-Surface Advection

Donor-Acceptor Method

Once the free surface has been reconstructed, the volume-of-fluid function F^s can be advected in time (in the remainder of this section the superscript s will be omitted for

presentational reasons). Basically, this is done using the donor-acceptor method that has been developed by Hirt and Nichols [35]. At each cell face the amount of fluid, based on the face area, the liquid velocity, and the time step, that is transported from the donor cell to the acceptor cell is computed (see figure 2.30 on the left). In cells not close to the

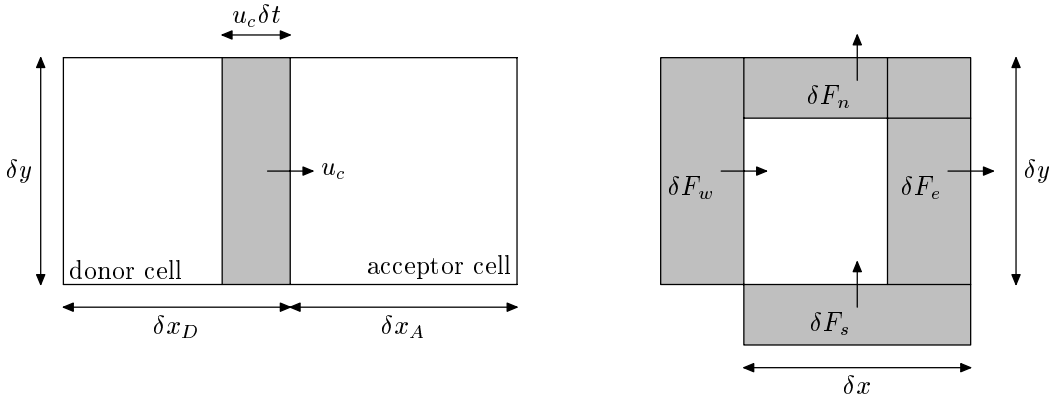

Figure 2.30: *Computation of the flux across a cell face using the donor-acceptor method.*

free surface, the flux δF_c across a cell face perpendicular to the x-direction is computed as

$$\delta F_c = u_c \delta t A_c^x \delta y, \tag{2.34}$$

where u_c is the normal velocity at the cell face, δt is the time step, and $A_c^x \delta y$ is the area of the part of the cell face that is open for flow. If for each cell face the flux has been computed (see figure 2.30 on the right), the VOF function is advected from time level n to $n+1$ by

$$F_c^b \delta x \delta y F_c^{n+1} = F_c^b \delta x \delta y F_c^n - \delta F_e - \delta F_n + \delta F_w + \delta F_s.$$

Although the basic principles of the donor-acceptor method, as described above, are simple, the computation of fluxes through cell faces near the free surface is more complicated since cells need not be completely filled with liquid here. Hence, the flux (2.34) may exceed the amount of fluid that is present in the donor cell. This is the cause of undershoots (below zero) and overshoots (above one) in the VOF function at the new time level. Hence, in the original VOF method, a flux near the free surface is computed as

$$|\delta F_c| = \min\left\{F_{AD}|u_c\delta t| + CF, F_D \delta x_D\right\}, \tag{2.35}$$

where

$$CF = \max\left\{(1 - F_{AD})\,|u_c\delta t| - (1 - F_D)\,\delta x_D, 0\right\}. \tag{2.36}$$

In these equations F_D and F_A denote the VOF values in the donor and acceptor cell respectively. The double subscript AD represents either the donor or the acceptor cell, depending on the orientation of the free surface relative to the direction of the flow. If the free surface is advected more or less normal to itself or if either the acceptor cell or the cell upstream of the donor cell is empty, then AD is set to A, otherwise AD is set to D. If the free surface is moving in tangential direction, setting AD to A would result in an unphysical steepening of the interface. The min operator in equation (2.35) prevents the fluxing of more fluid to the acceptor cell than the donor cell contains and

the max operator in (2.36) accounts for an additional flux if the amount of void exceeds the amount available. More details are explained in [35].

Local Height Function

An important drawback of the original method by Hirt and Nichols is the creation of "flotsam" and "jetsam" — small bits of fluid that get (unphysically) separated from the main body of fluid (see for example figure 2.35 and the references mentioned in section 1.4) — during the advection of the free surface. Further, this method may result in a considerable gain or loss in mass (despite the min and max limiters in equations (2.35) and (2.36)) since, at the end of every time cycle, VOF values below zero are reset to zero and VOF values above one are reset to one. Therefore, a method was developed that overcomes these problems (see [46, 63] for alternative methods for removing flotsam and jetsam). Since flotsam and jetsam is created near the free surface, the advection method of Hirt and Nichols is adapted in S cells. In these cells the fluid is transported using the local height function that already has been introduced for computing the curvature of the free surface (see section 2.3.6). The idea is that not the individual fluxes at the cell faces belonging to the S cell are transported, but the height of the free surface in the row or column corresponding to the S cell is updated. This is illustrated in figure 2.31. If the local

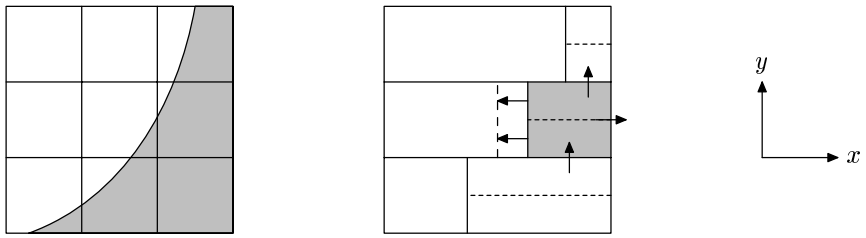

Figure 2.31: *Fluxes near the free surface are transported by changing the local height function.*

height function in the centre S cell in this figure is given by $S(x, y, t) = h(y, t) - x = 0$, then the advection equation (2.30) results in

$$\frac{\partial h}{\partial t} = v\frac{\partial h}{\partial y} - u,$$

where u and v are the velocity components in x- and y-direction respectively.

Youngs' Method

The limiters in equations (2.35) and (2.36) of the original VOF method are necessary to prevent severe undershoots and overshoots in the VOF function and to prevent the free surface from diffusing. The main reason for this is the characterisation of the interface to be either horizontal or vertical (in two dimensions). With the piecewise-linear interface reconstruction that has been explained in section 2.4.1, a more accurate method for computing fluxes across cell faces near the free surface can be applied. This method was introduced by Youngs [96] and is illustrated in figure 2.32. Instead of fluxing an amount

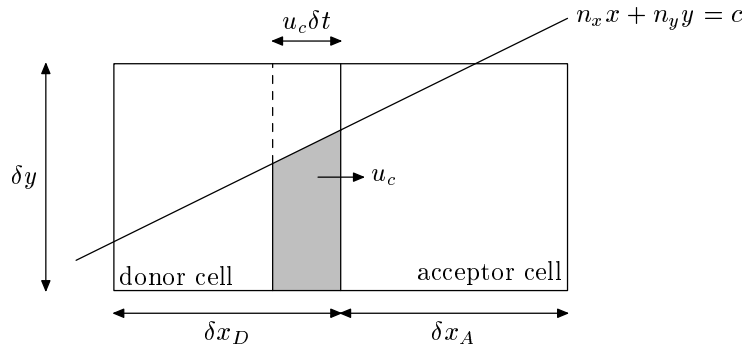

Figure 2.32: *Computation of the flux across a cell face using the piecewise-linear reconstruction of the interface.*

of $u_c \delta t A_c^x \delta y$, only the part of this volume that is inside the fluid ($n_x x + n_y y > c$) is fluxed (the shaded region in the figure). The region that has to be fluxed is computed with the same method that has been explained in the previous section; the volume integral is rewritten as a boundary integral along the boundary of the shaded region in such a way that the contribution of the free surface to this integral vanishes. The adaptation of the original VOF method to prevent the creation of flotsam and jetsam can also be applied in Youngs' method; the fluxes are computed as described above, however, the advection of the VOF function in S cells is performed using the local height function.

When the local height function is used for the advection of the free surface, the liquid mass is better conserved since fewer undershoots and overshoots in the VOF function occur. This important feature of the local height function is demonstrated in sections 2.5.1 and 2.5.2, where four advection methods — both the methods of Hirt and Nichols and of Youngs with and without the local height function — are compared with each other.

2.5 Results

2.5.1 Free-Surface Advection

An important part of the numerical model deals with the free-surface reconstruction and advection (see section 2.4). In this section some results regarding the advection of the free surface are presented. Four methods are compared with each other, namely Hirt and Nichols, Hirt and Nichols combined with the local height function, Youngs, and Youngs combined with the local height function (details of these methods have been discussed in section 2.4.2). The results in this section have been obtained with a prescribed velocity field; thus the Navier-Stokes equations are not solved. This makes it possible to test the free-surface reconstruction and advection only [42, 61].

Translation Test

First, some simple translation tests are performed. An initially circular or square body of fluid is placed in two different velocity fields. The first velocity field is aligned with the y-axis ($(u, v) = (0, 1)$). In the second velocity field the fluid is transported diagonally

through the Cartesian grid ($(u, v) = (2, 1)$). In all the simulations, the fluid is transported in 1000 time steps over a distance that is approximately five times the diameter of the fluid. The number of grid cells along the diameter of the fluid is equal to 15. The results are shown in figure 2.33. From this figure it is clear that Youngs' method is more accurate

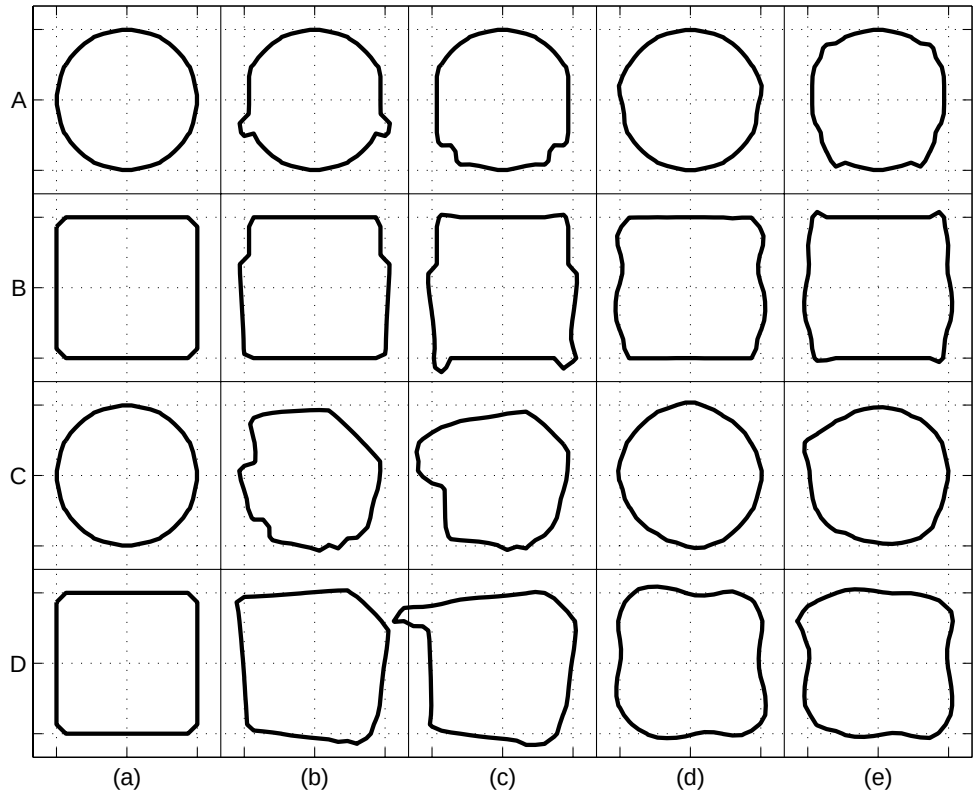

Figure 2.33: *Translation of a circle (rows A and C) and a square (rows B and D) in two different velocity fields* $(u, v) = (0, 1)$ *(rows A and B) and* $(u, v) = (2, 1)$ *(rows C and D). (a) Initial condition, (b) Hirt and Nichols without local height function, (c) Hirt and Nichols with local height function, (d) Youngs without local height function, and (e) Youngs with local height function.*

than the method of Hirt and Nichols. Especially in the diagonal translation of a circle, Youngs' method produces qualitatively better results. Further, the methods with the use of a local height function perform somewhat worse than the methods without its use.

Rotation Test

A more difficult test case is the solid-body rotation of a fluid region. Again, an initially circular and square body of fluid is considered. The distance between the centre of rotation and the centre of mass of the fluid is approximately twice the diameter of the fluid region. In about 600 time steps the fluid is rotated 360 degrees. The number of grid cells along the diameter of the fluid is equal to 15. The results are shown in figure 2.34. The small drops of fluid that appear in column (a) actually belong to column (b). This flotsam and jetsam is typical for the original method of Hirt and Nichols. Clearly, Youngs'

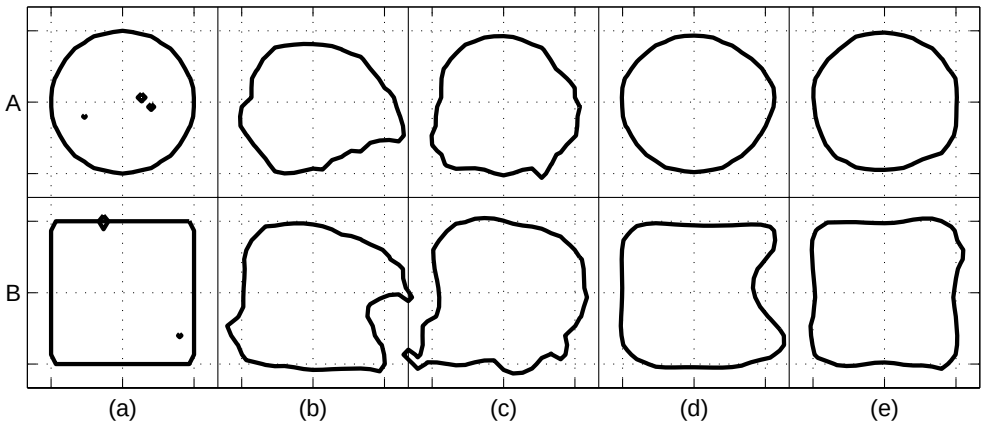

Figure 2.34: *Solid-body rotation of a circle (row A) and a square (row B). (a) Initial condition, (b) Hirt and Nichols without local height function, (c) Hirt and Nichols with local height function, (d) Youngs without local height function, and (e) Youngs with local height function.*

method produces better results than the method of Hirt and Nichols. However, also in Youngs' method the square region of fluid is quite distorted if the height function is not used. In the solid-body rotation the differences between the methods with and without the local height function are smaller than in the relatively simple translation tests. For the solid-body rotation of a square body of fluid, the use of a local height function even improves Youngs' method.

Single Vortex

The translation and rotation tests that have been discussed above are simple advection tests in the sense that (theoretically) the fluid topology does not change in time. Although these tests give some kind of guarantee that the advection method is correctly implemented, they do not give a definite indication of the capabilities of the method since in most practical applications the topology of the fluid changes considerably in time (*e.g.* breaking waves). Therefore, a more complex advection test, proposed by Rider and Kothe [61], is performed (see also [62]). A circle, with radius 0.15 and centred at $(0.50, 0.75)$, is placed in a square computational domain of unit dimensions. The circle is advected with a solenoidal velocity field $(u, v) = (-d\Phi/dy, d\Phi/dx)$, where

$$\Phi = \frac{1}{\pi} \sin^2(\pi x) \sin^2(\pi y) .$$

This velocity field causes the circular body of fluid to stretch and spiral around the centre of the domain as is shown in figure 2.35. The simulation is continued for T seconds. After $T/2$ seconds the flow field is reversed, such that at the end of the simulation, ideally, the fluid returns to its initial condition. In figure 2.35, snapshots of the simulation at $t = T/2$ and $t = T$ are shown for grids consisting of 60×60 and 100×100 cells and for simulation times $T = 2$ and $T = 4$. As for the translation and rotation tests, four different advection methods are studied. For the single vortex, the creation of flotsam and jetsam in the

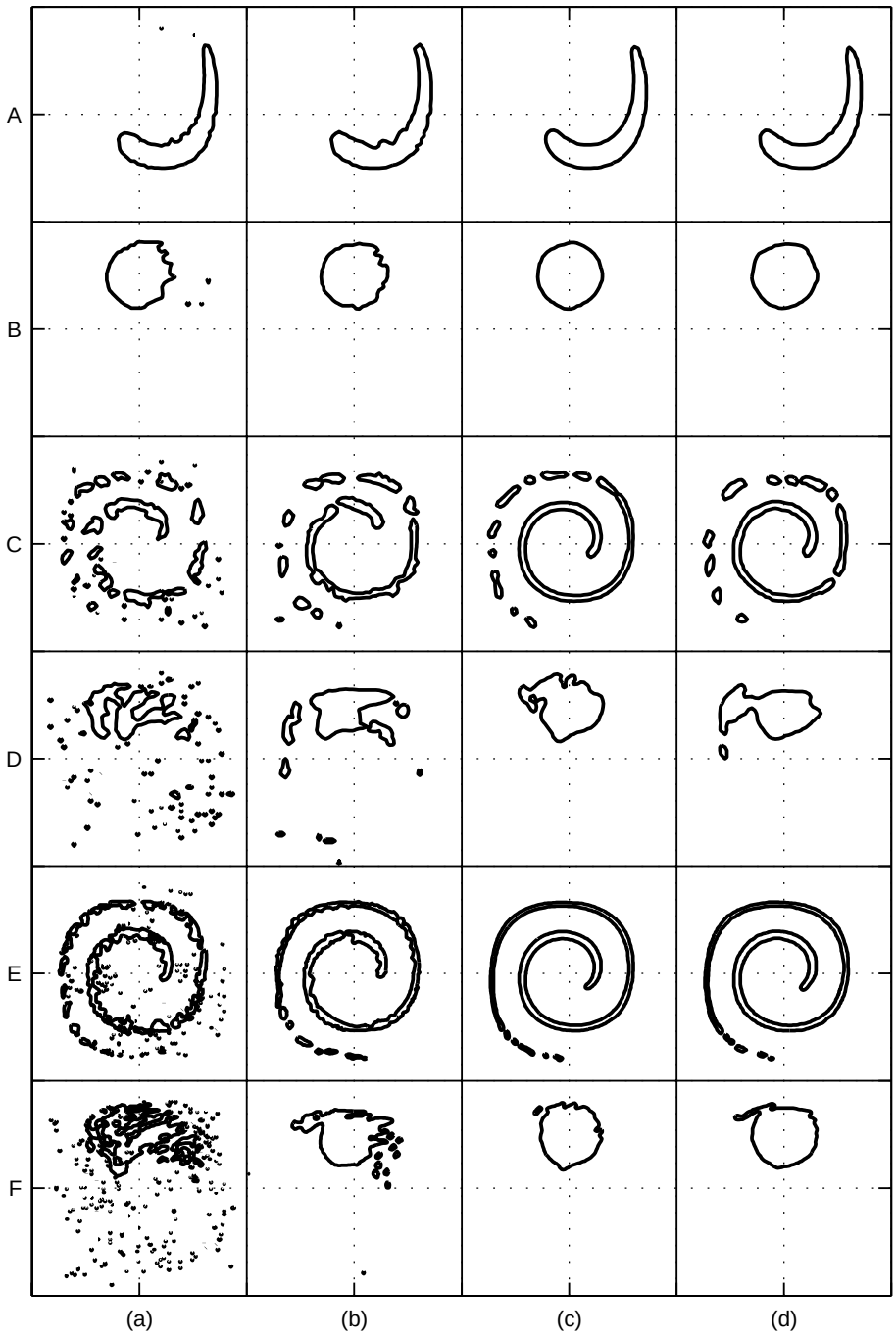

Figure 2.35: *Advection of a circular body of fluid placed in a single-vortex flow field. The maximum simulation time is equal to* $T = 4$ *except for rows A and B, where* $T = 2$*. Snapshots are shown at time* $T/2$ *(rows A, C, and E) and at time* T *(rows B, D, and F). The computational grid consists of* 60×60 *cells (rows A, B, C, and D) or* 100×100 *cells (rows E and F). (a) Hirt and Nichols without local height function, (b) Hirt and Nichols with local height function, (c) Youngs without local height function, and (d) Youngs with local height function.*

original method of Hirt and Nichols becomes very apparent. If a local height function is used in combination with this method, then the fluid remains much more coherent. Qualitatively, Youngs' method reconstructs the circle better than the method of Hirt and Nichols. Indeed, the original method of Youngs does not suffer from the creation of flotsam and jetsam. Hence, the use of a local height function does not affect Youngs' method much. Especially when the simulation time is long and a fine grid is used, Youngs' method reconstructs the circle better than the method of Hirt and Nichols. On the coarser grid the circle is heavily distorted for both the methods.

A more quantitative way of comparing the various advection methods is by examining the error in the VOF function F at time $t = T$ (see also [30, 61]). In table 2.1 the discrete error norm

$$\frac{1}{N} \sum_{i,j} \left| F_{i,j}\left(t = T\right) - F_{i,j}\left(t = 0\right) \right|,$$

where N is the number of grid cells and the summation is over the grid cells, is presented for various values of N and T. For all the simulations corresponding to this table, a

T	grid	H&N −	H&N +	Youngs −	Youngs +
0.5	32×32	5.58×10^{-3}	6.11×10^{-3}	4.19×10^{-3}	4.01×10^{-3}
0.5	64×64	3.21×10^{-3}	3.64×10^{-3}	2.08×10^{-3}	2.33×10^{-3}
0.5	128×128	2.64×10^{-3}	2.14×10^{-3}	1.06×10^{-3}	1.34×10^{-3}
2.0	32×32	1.90×10^{-2}	1.07×10^{-2}	6.04×10^{-3}	7.23×10^{-3}
2.0	64×64	1.02×10^{-2}	5.13×10^{-3}	2.97×10^{-3}	3.19×10^{-3}
2.0	128×128	7.90×10^{-3}	3.31×10^{-3}	1.93×10^{-3}	1.63×10^{-3}
8.0	32×32	1.04×10^{-1}	8.39×10^{-2}	5.95×10^{-2}	1.10×10^{-1}
8.0	64×64	7.96×10^{-2}	6.05×10^{-2}	2.00×10^{-2}	4.49×10^{-2}
8.0	128×128	6.76×10^{-2}	3.17×10^{-2}	8.25×10^{-3}	8.02×10^{-3}

Table 2.1: *Error norm in the VOF function for the single-vortex advection test. Comparison of four advection methods: Hirt and Nichols (H&N) and Youngs, both without (−) and with (+) the use of a local height function. The CFL number is constant for all simulations.*

constant CFL number of 0.25 has been used. This means that on finer grids a smaller time step was chosen. Hence, the results in the table combine a grid-refinement study and a time-step analysis. Clearly, for all advection methods the error in the VOF function decreases if the grid is refined, although the number of time steps increases upon grid refinement. Except for one case, Youngs' method is superior compared to the method of Hirt and Nichols. Further, in almost all cases, the use of a local height function improves the original method of Hirt and Nichols; only when the number of time steps is small ($T = 0.5$, grids 32×32 and 64×64), the height function worsens the results. Moreover, the improvement in the method of Hirt and Nichols by using the height function is better on fine grids. For the method of Youngs the use of a local height function is less beneficiary; only when the number of time steps is very large or very small, the local height function improves the original method of Youngs. Moreover, the differences in the error between Youngs' method with and without a local height function is generally smaller than the differences between the method of Hirt and Nichols with and without the

height function. In table 2.2 the order of convergence upon grid refinement and time-step reduction is studied. Although the variation in the ratios of the error norm is quite high,

T	grid	H&N −	H&N +	Youngs −	Youngs +
0.5	$32^2/64^2$	1.74	1.68	2.02	1.72
0.5	$64^2/128^2$	1.22	1.70	1.96	1.75
2.0	$32^2/64^2$	1.87	2.09	2.03	2.27
2.0	$64^2/128^2$	1.29	1.55	1.54	1.95
8.0	$32^2/64^2$	1.31	1.39	2.97	2.44
8.0	$64^2/128^2$	1.18	1.91	2.43	5.60

Table 2.2: *Order of convergence of error norm in the VOF function for the single-vortex advection test. Comparison of four advection methods: Hirt and Nichols (H&N) and Youngs, both without (−) and with (+) the use of a local height function.*

it is concluded that the original methods of Hirt and Nichols and of Youngs are first-order and second-order accurate respectively. This conclusion is drawn not only because of the results in table 2.2, but also because of the way the methods are constructed; while Hirt and Nichols reconstruct the interface by approximating it with horizontal and vertical line segments, the method of Youngs exploits a piecewise-linear reconstruction. The order of convergence of the former method appears to increase if it is combined with a local height function. For Youngs' method no definite conclusion can be made on the change of the order of convergence if a height function is used.

An important aspect of an advection method is the capability of conserving mass. Since undershoots and overshoots in the VOF function are cut off at the end of every time cycle (to fulfil the condition $0 \leq F^s \leq F^b$ at the beginning of a new time cycle), a considerable amount of mass may be gained or lost during a simulation. The local height

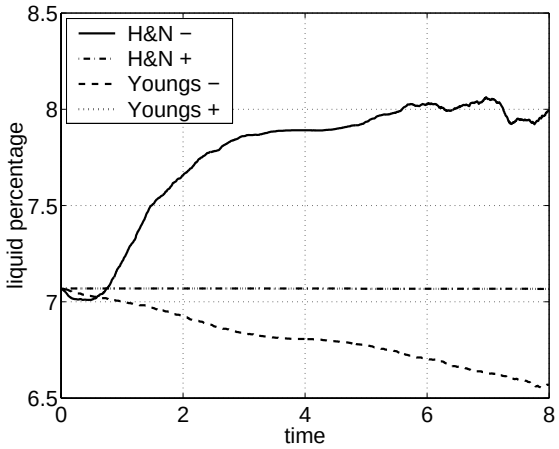

Figure 2.36: *Liquid percentage as a function of time for the single-vortex advection test (with $T = 8$ and a grid of 128×128 cells). Comparison of four advection methods: Hirt and Nichols (H&N) and Youngs, both without (−) and with (+) the use of a local height function.*

function was introduced to prevent large undershoots and overshoots in the VOF function. Hence, an advection method combined with a local height function for transporting fluxes across cell faces is expected to conserve mass better than the same advection method without the local height function. This is confirmed in figure 2.36, where the liquid percentage for the single-vortex simulation is shown as a function of time for the four different advection methods that have been studied in this section. Clearly, in the original method of Hirt and Nichols mass is gained due to undershoots in the VOF function and mass is lost in the original method of Youngs due to overshoots. If a local height function is used in combination with these methods, the liquid percentage remains constant in time.

2.5.2 Static Drop and Drop Oscillation

In this section results of simulation of fluid flow in a zero-gravity environment are presented. The liquid is not in contact with the solid body, whence only the numerical model for surface tension is validated (the numerical model for the contact angle is discussed in the next section). First, a static, circular liquid drop is studied. Then, the transformation of a square liquid drop into a circular drop under the influence of surface tension is discussed.

Static Drop

A circular body of fluid (in two dimensions), that is subjected to surface-tension forces only, is in static equilibrium. However, since the analytical VOF field need not correspond to the discrete VOF field of the steady-state solution and because of numerical errors in the boundary conditions applied at the free surface, spurious currents may occur [46, 60]. In the CSF method of Brackbill *et al.* [5], where also the motion of the surrounding air is resolved, these spurious currents are quite dramatic as they lead to an increase of the kinetic energy as has been demonstrated by Rudman [63]. In the left of figure 2.37, the evolution of the kinetic energy for the static drop is shown for the method of Hirt and Nichols and for Youngs' method, both in combination with a local height function for advecting fluid near the free surface. Clearly, in Youngs' method, the kinetic energy does not decrease; small oscillations at the free surface grow in time and the drop starts drifting away from its initial position. Apparently, this method is not able to damp small disturbances. The reason for this lies in the fact that the advection of the free surface is completely based on the linear reconstruction of the free surface in S cells. Due to small numerical errors, the reconstructed normal at the free surface is not symmetrical around the drop, whence the computed fluxes are not symmetrical. Hence, the distribution of the VOF field is not symmetrical at the next time step, which, in effect, results in an asymmetrical computation of the normal at the free surface, and so forth (see also [30], where it was concluded that the interface reconstruction is crucial for the overall performance of an advection algorithm). These problems do not occur in the method of Hirt and Nichols. Indeed, in this method the free surface is classified as being either horizontal or vertical, which is not sensitive to small asymmetries in the VOF field. For this method the kinetic energy tends to zero as is shown in the left of figure 2.37 for two different Ohnesorge numbers.

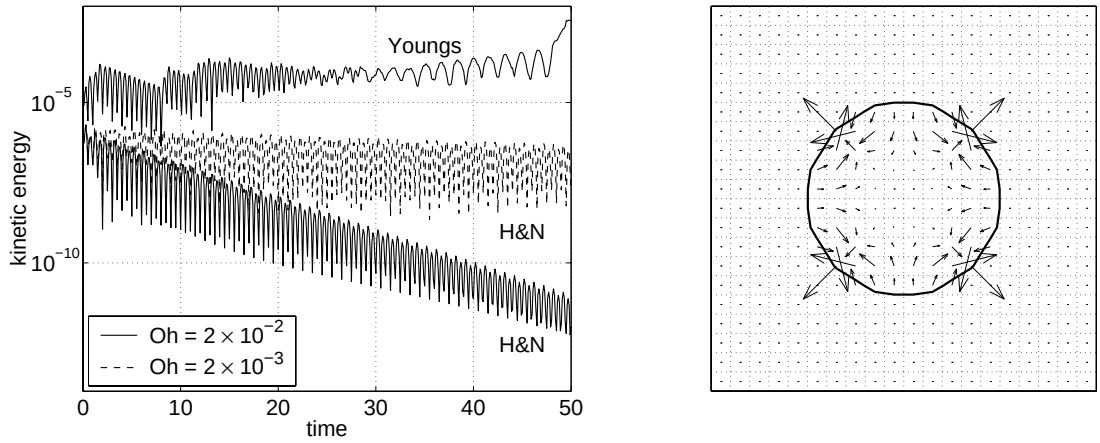

Figure 2.37: *Left: Evolution of the kinetic energy for the two-dimensional static drop for* $Oh = 2 \times 10^{-2}$ *and* $Oh = 2 \times 10^{-3}$*. Right: Snapshot of the free surface and vector plot of the velocity field after 10 time steps.*

Although in the method of Hirt and Nichols the kinetic energy decreases in time, spurious currents are present in this method as is shown in the right of figure 2.37. In this figure a snapshot of the free surface and a vector plot of the velocity field is shown after 10 time steps, clearly demonstrating the presence of spurious currents. For this simulation a circle with radius 0.25, centred in a square domain of unit dimensions, has been studied. The computational grid consisted of 20×20 cells and a time step of $\delta t = 10^{-2}$ was used. The Ohnesorge number $Oh = \mu/\sqrt{\rho\sigma R}$, indicating the ratio between viscous and surface-tension forces (R is a characteristic length scale, *e.g.* the radius of the circular body of fluid), was set to $Oh = 2 \times 10^{-2}$ and $Oh = 2 \times 10^{-3}$. Clearly, since Oh is proportional to the viscosity of the fluid, the kinetic energy is damped more for a higher value of Oh.

In figure 2.37 the free surface appears somewhat rippled. This is due to the post-processing software; the free surface is drawn as a contour line of the VOF field at a value 0.5. Although being more computationally expensive, a better way to show the

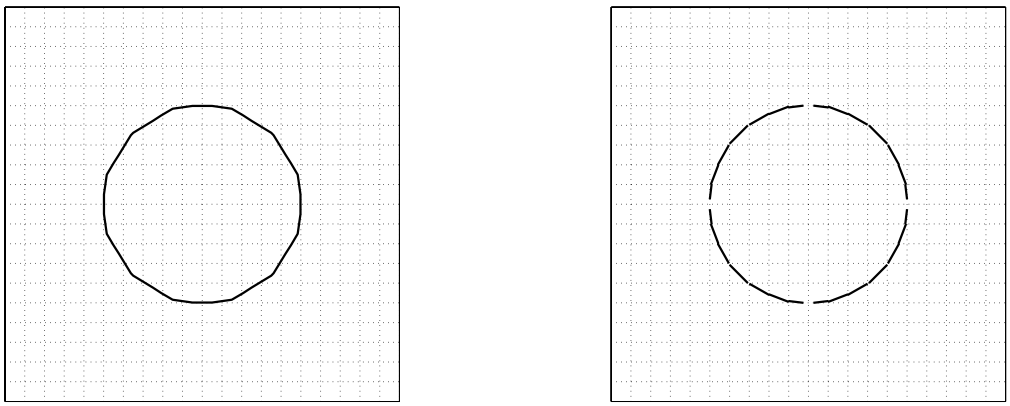

Figure 2.38: *Snapshot of the free surface drawn as a contour line of the VOF field (left) and as a piecewise-linear reconstruction (right).*

free surface is by plotting the reconstructed free surface (see section 2.4.1): in every computational cell that is cut by the free surface a line segment is drawn. The difference between the two methods is shown in figure 2.38. For convenience, however, most of the snapshots in the remaining part of this thesis show the free surface as a contour line of the VOF field.

The simulation of a static drop has been repeated in three dimensions. For this simulation the evolution of kinetic energy for two different Ohnesorge numbers is shown in figure 2.39. Although the kinetic energy decreases not so regular as in two dimensions, the simulations are stable: small disturbances are damped.

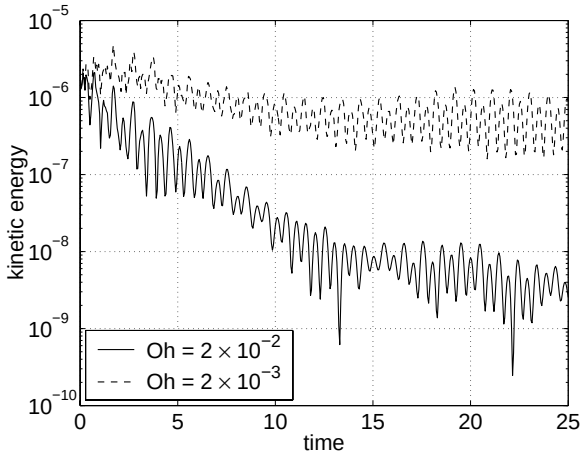

Figure 2.39: *Evolution of the kinetic energy for the three-dimensional static drop for* $Oh = 2 \times 10^{-2}$ *and* $Oh = 2 \times 10^{-3}$.

Drop Oscillation

If the initial shape of the fluid is not circular but square and the fluid is subjected to surface tension, then the fluid is not in equilibrium. Indeed, the curvature in the corners of the square is infinite and along the edges the curvature is equal to zero. Hence, the surface-tension force pulls the fluid from the corners towards the centre. Because of mass conservation and symmetry, the fluid at the centres of the edges starts flowing outward. After some time, the free surface resembles the initial configuration rotated about 45 degrees, whence the cycle starts over again. Of course, because of viscosity, the overshoots become smaller and smaller, such that, eventually, the fluid settles itself in a circular equilibrium position (see also [5, 55, 70, 72]).

Snapshots of the evolution of a square drop into a circular drop are shown in figure 2.40. For this simulation a square drop of dimensions $\frac{1}{2} \times \frac{1}{2}$ is placed in the centre of a square domain of unit dimensions. The grid consists of 40×40 cells and the time step is set to $\delta t = 10^{-3}$. The Ohnesorge number for this simulation is equal to $Oh = 0.1$. Note that in figure 2.40 the velocities are scaled with the largest velocity; in fact, the kinetic energy decreases to machine precision during the simulation.

This simulation has been performed both with the method of Hirt and Nichols and the method of Youngs. The snapshots in figure 2.40 correspond to the former method. For

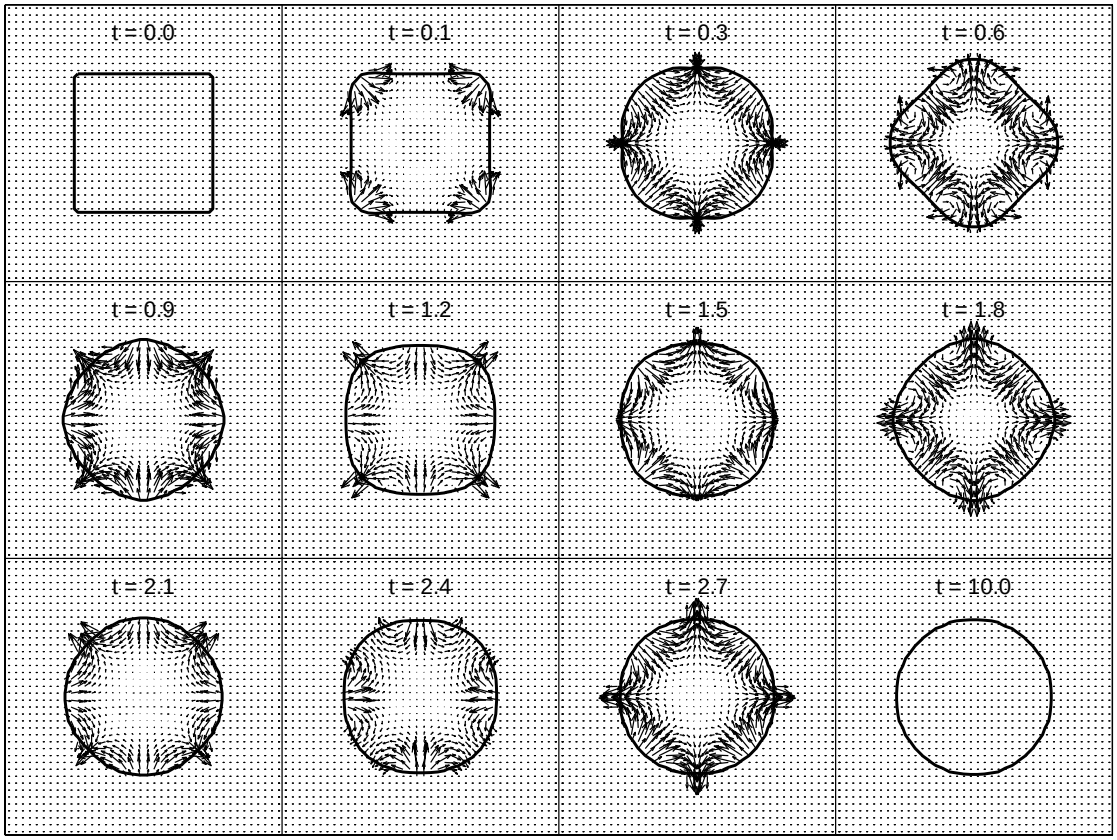

Figure 2.40: *Snapshots of the evolution of a square drop into a circular drop (computed with the method of Hirt and Nichols combined with a local height function). Shown are the shape of the free surface (solid lines) and the velocity field (arrows). In each snapshot the velocity is scaled with the largest velocity.*

this method the centre of mass of the liquid drop remains in the centre of the computational domain throughout the entire simulation. Small asymmetries in the VOF field and the velocity field (see for example the snapshot at time $t = 1.5$, where the relatively large velocity in the north has no counterpart in the east, south, or west) are damped. In the method of Youngs, the kinetic energy does not decrease to machine precision because the drop starts drifting away from the centre of the computational domain and, eventually, impacts with the solid boundary; see also the discussion for the static drop above.

Apart from the snapshots, also the diameter of the liquid drop along the horizontal centreline is studied. This is done in two and three dimensions on grids consisting of either 20 or 40 cells in every Cartesian direction. The results are shown in figure 2.41. Since the initial amount of fluid is known (and is constant in time because the local height function is used) and the shape of the free surface in the steady-state solution is either circular or spherical, the diameter of the liquid drop in the steady state can be compared to theory easily. This steady-state diameter is plotted in figure 2.41 as a dash-dotted, horizontal line. For the coarse-grid solutions the difference between the numerical and the theoretical steady-state diameter is 0.60% and 0.77% in two and three dimensions respectively. On the fine grid these errors equal 0.18% and 0.33% respectively.

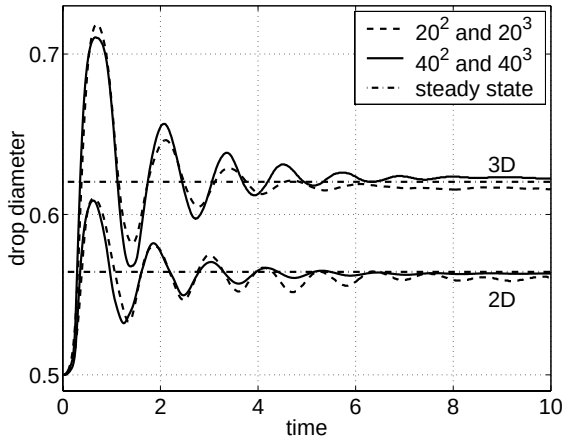

Figure 2.41: *Diameter of the liquid drop as a function of time in two and three dimensions for various grids. The dash-dotted lines indicate the theoretical drop diameters of the steady-state solution.*

2.5.3 Two-Dimensional Wall Adhesion in Square Cavity

In this section capillary flow in a (two-dimensional) square cavity is studied. Initially, the lower half of the cavity is filled with liquid. Surface tension is the only force present. A static contact angle θ is prescribed. The discrepancy between this contact angle and the initial angle between the free surface and the solid boundary drives the flow. Due to surface tension, the free surface will settle itself into an equilibrium configuration in which the mean curvature of the free surface is constant and the angle between the free surface and the solid boundary is equal to the prescribed contact angle; thus the shape of the free surface in the steady-state situation will be circular (see also [5, 55]).

Grid-Refinement Study

First, wall adhesion in a square container is simulated with $\theta = 45$ degrees. In figure 2.42 snapshots of the free surface are shown on various grids. For these simulations the Ohnesorge number was set to $Oh = 10^{-2}$. The snapshots are taken at time $t = 80$ when the fluid has reached a steady state (by this time the kinetic energy is close to machine precision). The bold line segments in figure 2.42 represent the reconstructed free surface and the thin, solid line shows the theoretical shape of the free surface. On the coarse grid of 20×20 cells a clear difference between simulation and theory is present; the grid is too coarse to predict the correct steady-state configuration of the free surface. On the finer grids of 40×40 and 80×80 cells the agreement is very good. A more quantitative comparison between simulation and theory is made in figure 2.43. In this figure, the liquid height at the wall and at the centreline of the cavity is plotted as a function of time and the theoretical liquid height in the steady-state solution is shown as a horizontal line. Again, the oscillation around the steady-state solution, which is typical for flows driven by surface tension, is present. From figure 2.43 it is clear that the error in the liquid height is larger at the solid wall than at the centreline of the cavity. The error in the steady-state liquid height at the wall is equal to 8.0%, 3.5%, and 1.5% for the

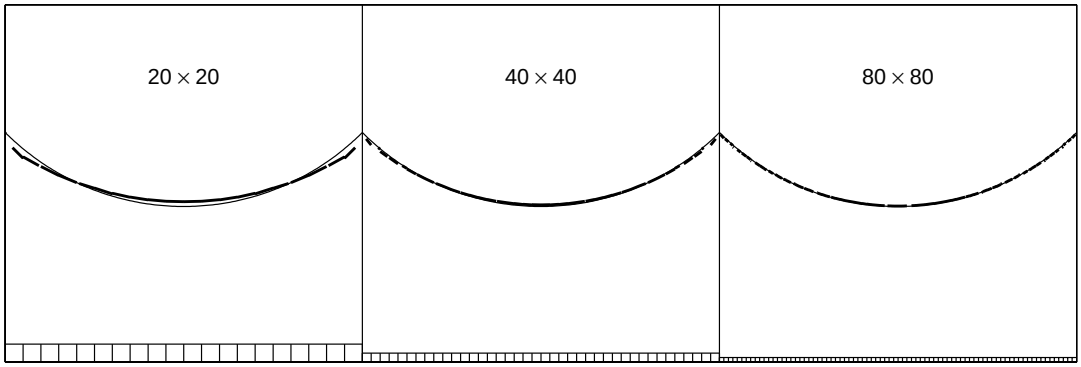

Figure 2.42: *Snapshots of the free surface in the steady state solution ($\theta = 45$) on grids of 20^2, 40^2, and 80^2 cells (a small part of the computational grid is shown in the bottom of the cavity). The bold line segments represent the reconstructed free surface. The solid thin line shows the theoretical steady state solution.*

coarse, medium, and fine grid respectively. Part of this error is due to the fact that, in the simulation, the liquid height at the wall is computed by adding the VOF values in the first (or last) column of the computational grid. This also explains the underprediction of the liquid height in the simulation. If this effect is taken into account, then the error in the liquid height on the coarse, medium, and fine grid is equal to 4.4%, 1.7%, and 0.54% respectively.

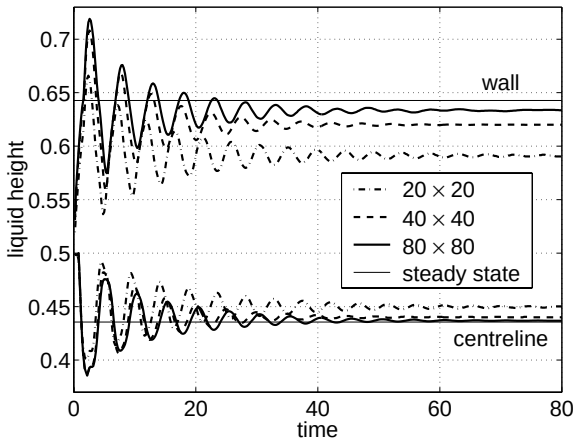

Figure 2.43: *Liquid height (for the simulation of wall adhesion in a square cavity ($\theta = 45$ and $Oh = 10^{-2}$)) at the wall and at the centreline as a function of time for grids consisting of 20^2, 40^2, and 80^2 cells.*

Variation of the Contact Angle

For checking the dependence of the free-surface shape on the contact angle, the same simulation as above is repeated with different contact angles. The steady-state solution

of the free surface is shown in figure 2.44 for $\theta = 0$, 30, 60, 120, 150, and 180 degrees. Since the results on the medium grid gave adequately accurate results for $\theta = 45$, for all the simulations in this figure the grid consists of 40×40 cells. For post-processing reasons, the free surface is plot as a contour line of the VOF field at a value 0.5; hence the ripples in the free surface. Clearly, the method is capable of handling contact angles throughout the entire spectrum from 0 to 180 degrees; the agreement between the computed and the theoretical free-surface shape is very good.

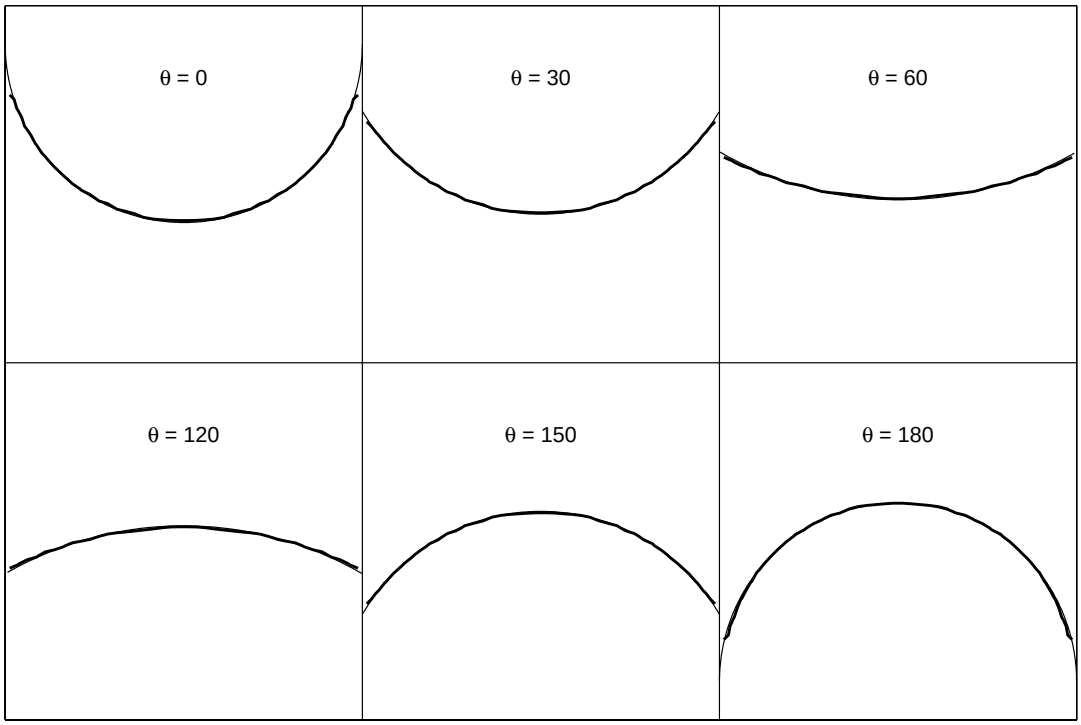

Figure 2.44: *Snapshots of the free surface in the steady-state solution for various contact angles. The bold lines show results of the simulation and the thin lines show theoretical results.*

Hirt and Nichols versus Youngs

Based on experiences in section 2.5.2, all the simulations in this section have been performed with the method of Hirt and Nichols in combination with a local height function. As a final comparison between the two methods for free-surface advection, the simulation in this section with contact angle $\theta = 30$ is repeated for both methods without the use of a local height function. Results of the simulation with the method of Hirt and Nichols are not presented here since these are almost identical to the ones presented before, where this method was combined with a local height function. This is due to the fact that, in simulations without massive change in topology of the free surface and moderate velocity gradients (like the simulations in this section), the original method of Hirt and Nichols gives similar results as the method extended with a local height function. The only difference is the slight increase or decrease of mass (for the simulations in this section in the

order of 0.01%) due to small undershoots and overshoots in the VOF function.

For the original method of Youngs the results are quite dramatic. A few snapshots of the free surface for the simulation with contact angle $\theta = 30$ are shown in figure 2.45. Contrary to previous results in this section, the fluid reaches the top of the cavity (also

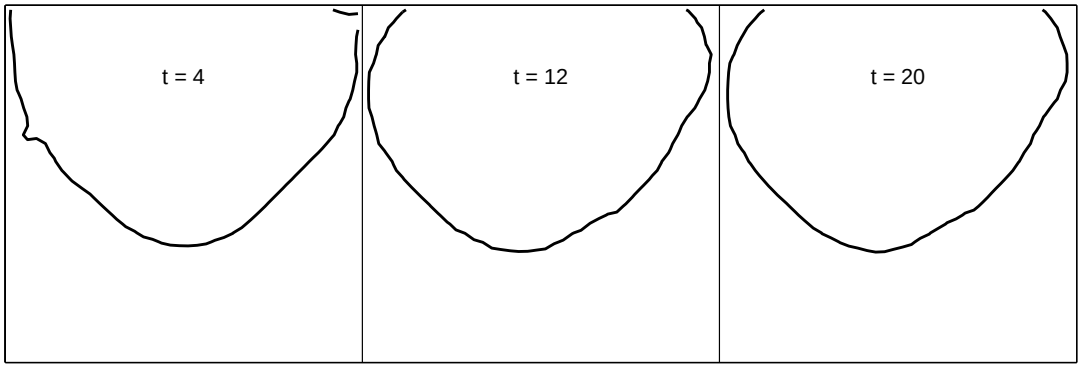

Figure 2.45: *Snapshots of the free surface for the method of Youngs without a local height function. The contact angle is equal to* $\theta = 30$ *and the number of grid cells is equal to* 40×40.

when the value of the liquid viscosity is increased, fluid flows towards the top). Further, it is clear that symmetry is broken quite early in the simulation. It is concluded that the original method of Youngs, although it gives accurate results in simulations where the velocity field is prescribed (see section 2.5.1), gives worse results than the method of Hirt and Nichols in simulations with velocity fields that are computed from the Navier-Stokes equations.

If a local height function is used in combination with Youngs' method, then the results improve a lot; the fluid stays coherent throughout the simulation and does not reach the top of the flow domain. However, contrary to results of the method of Hirt and Nichols, the results are not symmetrical. Moreover, in Youngs' method, the kinetic energy does not decrease to machine precision as is shown in figure 2.46.

Since the results of simulations, where the velocity field is solved from the Navier-Stokes equations, show that the method of Hirt and Nichols combined with a local height function for advecting the free surface gives the best results, all the simulations in the remaining part of this thesis are performed with this method.

2.5.4 Two-Dimensional Wall Adhesion in Circular Cavity

All the simulations in this chapter have been performed in simple geometries, *i.e.* in geometries of which the boundary coincides with Cartesian grid lines. These simple geometries were chosen to be able to focus the validation on the numerical models for surface tension, contact angle, and free-surface advection. In this section, for validating the treatment of cut cells, wall adhesion is studied in a circular cavity. The circle has a radius equal to unity. Initially, the lower half of the flow domain is filled with liquid. The contact angle is set to $\theta = 30$. Together with surface tension, this contact angle drives the flow.

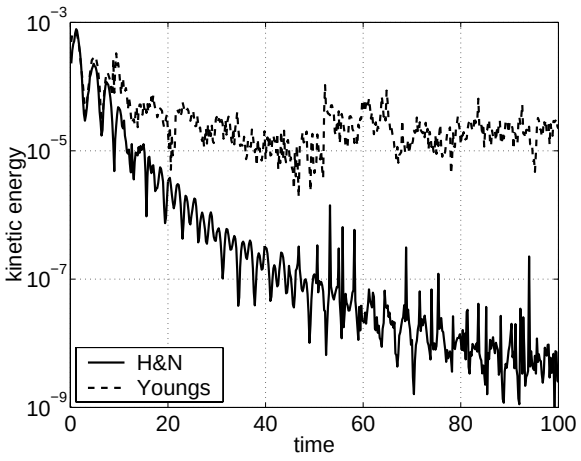

Figure 2.46: *Evolution of the kinetic energy in the wall-adhesion simulation ($\theta = 30$) for the method of Hirt and Nichols (H&N) and for Youngs' method, both in combination with a local height function.*

Grid-Refinement Study

In figure 2.47 snapshots (at time $t = 100$, when a steady state was reached) of the free surface are shown for grids consisting of 20×20, 40×40, and 80×80 cells. The Ohnesorge number for these simulations, based on the radius of the circular flow domain, was equal to $Oh = 10^{-2}$. In this figure, bold lines show results from the simulation, while thin lines correspond to the theoretical steady-state solution. As with the square cavity in figure 2.42, the grid consisting of 20×20 cells is too coarse to accurately resolve the liquid dynamics. The somewhat finer grid with 40×40 cells gives adequate results. The difference between theory and simulation is hardly visible on the finest grid with 80×80 cells.

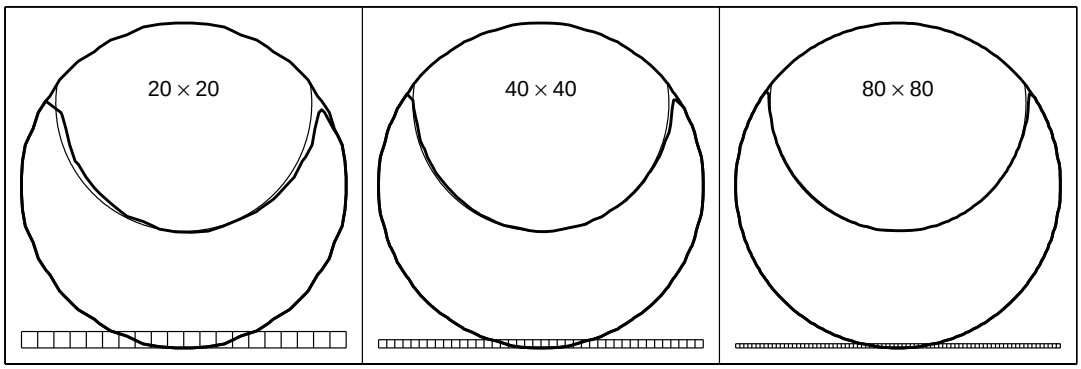

Figure 2.47: *Snapshots of the free surface in the steady-state solution ($\theta = 30$) on grids of 20^2, 40^2, and 80^2 cells (a small part of the computational grid is shown in the bottom of the flow domain). Bold lines show results from the simulation, thin lines show the theoretical steady-state solution.*

Staircase Geometry

To show the effect of taking into account cut cells instead of using a staircase discretisation of complex geometries, the simulation of wall adhesion in a circular cavity is repeated with a staircase approximation of the circular flow domain. This means that computational cells of which the centre lies outside the circular flow domain are B cells with a volume aperture $F^b = 0$. Volume apertures of interior cells are equal to $F^b = 1$. These cells are, depending on the presence of liquid, F, S, or E cells. For this simulation a grid with 80×80 cells was used. A snapshot of the free surface at time $t = 150$ is shown in the left of figure 2.48. Although the grid is quite fine, the free surface gets stuck in a corner of

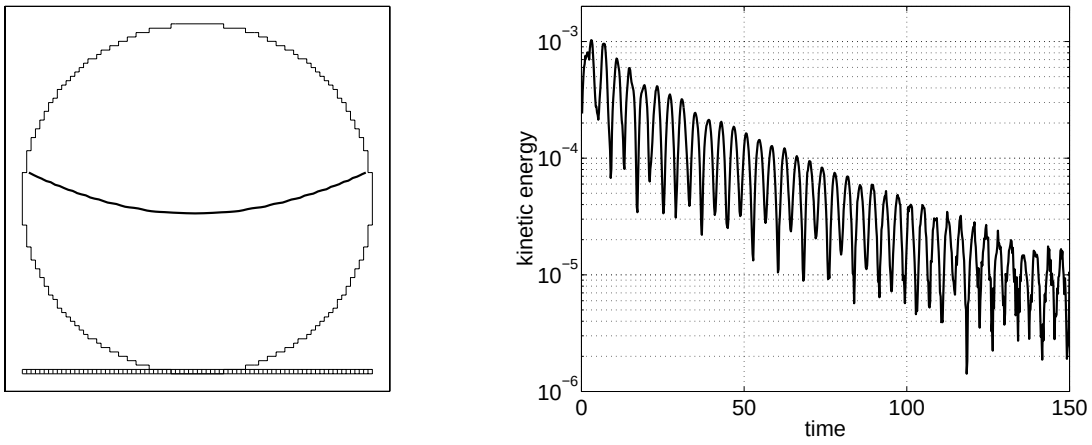

Figure 2.48: *Wall adhesion ($\theta = 30$) in a staircase approximation of the circular cavity. Left: Snapshot of the free surface at time $t = 150$. Right: Kinetic energy as a function of time.*

the staircase-shaped geometry. In order to show that this is not a temporary situation, the evolution of kinetic energy for this simulation is shown in the right of figure 2.48. Clearly, the configuration of the free surface at time $t = 150$ is almost identical to the steady state. This is confirmed by the circular shape of the free surface. Note that this solution is not in conflict with the prescription of a static contact angle of 30 degrees since in a non-smooth corner the contact angle can not be defined properly.

2.5.5 Three-Dimensional Wall Adhesion in Spherical Cavity

As a final example, wall adhesion in a (three-dimensional) spherical cavity is simulated. For this, a grid consisting of $40 \times 40 \times 40$ cells was used and the time step was set to $\delta t = 5 \times 10^{-4}$. Like in the previous simulations, the Ohnesorge number equaled $Oh = 10^{-2}$ and a contact angle $\theta = 30$ was prescribed. In figure 2.49 a three-dimensional snapshot of the steady-state solution is shown. To be able to see the free-surface shape, one quarter of the fluid is not drawn. Clearly, the free surface is spherical as theory predicts. However, it is not perfectly axisymmetric as is best seen at the contact line. The contact line is smooth in the direction of the Cartesian axes, while in the diagonal directions the free surface shows some ripples. This is due to the estimation of the contact line (see section 2.3.6), which is more difficult if the solid boundary is not aligned with a Cartesian direction.

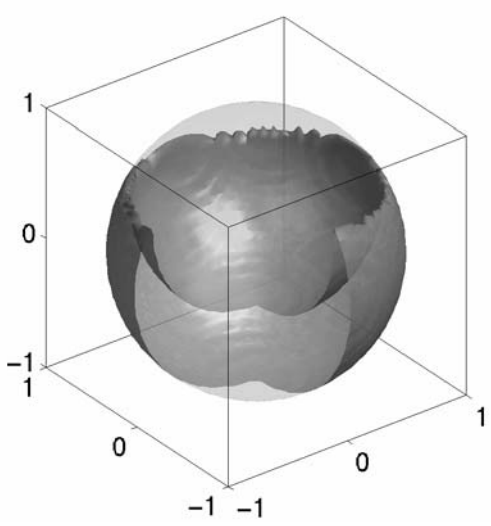

Figure 2.49: *Wall adhesion in a spherical cavity ($\theta = 30$). Snapshot of the steady-state solution. One quarter of the free surface has been removed to be able to see the interior free-surface shape.*