

Semi-Lagrangian VOF Advection Method for *FLOW-3D*[®]

Michael R. Barkhudarov

Flow Science, Inc.

July 2003

1. Introduction

A new VOF advection method based on a 3-D reconstruction of the fluid interface has been developed and implemented in *FLOW-3D*[®] Version 8.2. The Volume-of-Fluid (VOF) function is moved in one step, without resorting to an operator splitting technique, which gives the present method increased accuracy when the flow is not aligned with a coordinate direction.

The existing VOF advection method in *FLOW-3D*[®] (hereinafter called the *standard* method) is based on the donor-acceptor approach first introduced by Hirt and Nichols [1]. Numerous enhancements have been made to the original algorithm to improve its accuracy and stability in complex one- and two- fluid flows with sharp interfaces¹.

The standard method uses operator splitting and old time-level values of the VOF function to compute fluxes in three coordinate directions. The approach creates a possibility of overfilling or over-emptying computational cells when volume fluxes are significant in all three directions and the time step size is close to the local Courant stability limit.

The new advection method has been developed to alleviate these deficiencies of the standard algorithm. The fluid interface is reconstructed in 3D using a piecewise linear representation, where the interface is assumed to be planar in each control volume (or cell) containing the interface. The fluid volume bounded by the interface and cell faces is then moved according to the local velocity vector in a Lagrangian manner. Finally, the advected volume is overlaid back onto the Eulerian grid to obtain the new values of the fraction-of-fluid function. This combination of the Lagrangian and Eulerian methodology gives the new method its name (similar approaches have been used to approximate advection terms, for example, by Colella [2], Puckett *et al* [3], and Pilliod and Puckett [4]). The new option is activated in the code by setting IFVOF=5.

2. Standard VOF method

The fluid fraction function F is defined to be equal to 1.0 in the fluid and 0.0 outside fluid, *i.e.* in the void. Averaged over a control volume, the value of F will be within the segment [0.0, 1.0], as shown in Fig. 1. The volume of fluid in a cell is

$$Fvol = FV_f dx dy dz \quad (1)$$

¹ The standard method is invoked by setting IFVOF=4 in the *FLOW-3D*[®] input file, which is the default setting for one-fluid problems. For two-fluid problems with sharp interface the default value is 3.

The kinematic equation for the VOF function F in the absence of mass sources is

$$V_f \frac{\partial F}{\partial t} + \nabla \cdot (\mathbf{A}UF) = 0 \tag{2}$$

where V_f and $\mathbf{A}=(A_x, A_y, A_z)$ are the volume and area fractions describing the geometrical constraints of the flow, $\mathbf{U}=(U, V, W)$ is the flow velocity, and

$$\mathbf{A}U = (A_x u, A_y v, A_z w) \tag{3}$$

The numerical solution of Eq. (2) must prevent unphysical distortion of the interface and preserve its sharpness. The latter requirement means that the interface between fluid and void must be no more than one cell wide.

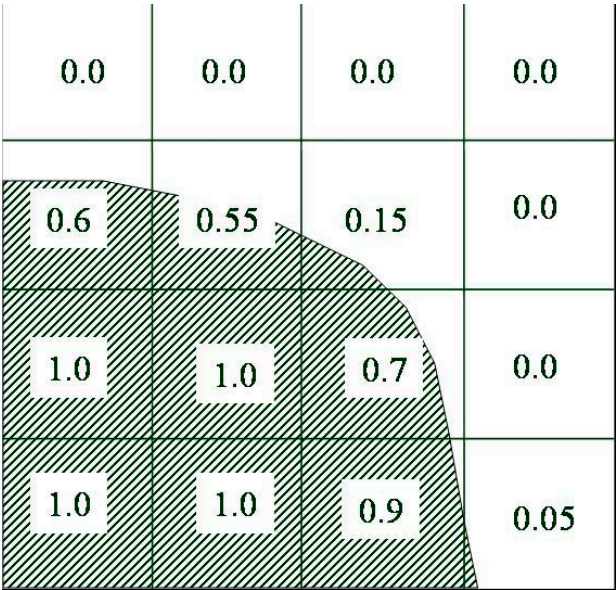


Figure 1. Typical values of the VOF function near free surface.

The most robust, accurate and commonly used way to achieve both goals in fixed Eulerian grids is to compute volumetric fluxes by geometrically reconstructing the interface using the values of the VOF function in and around a given control volume. Most existing methods that use this approach differ in the way the shape of the interface is approximated, resulting, among other names, in the SLIC, PLIC, and Least Squares methods (see, for example, Pilliod and Puckett [4]). Such methods have good volume conservation properties.

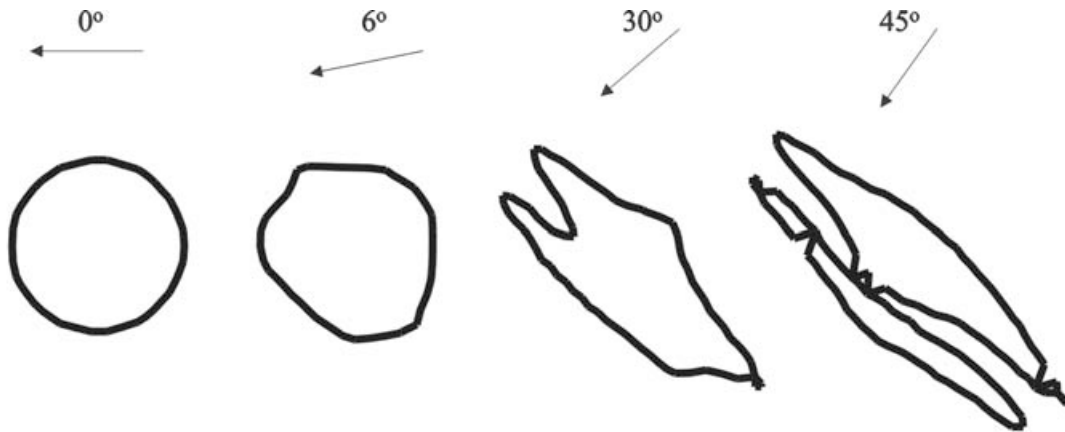


Figure 2. The final shapes of an initially round droplet after advection in four directions using the standard VOF method. An arrow above the each picture indicates the direction of the flow and the angle to the horizontal coordinate direction.

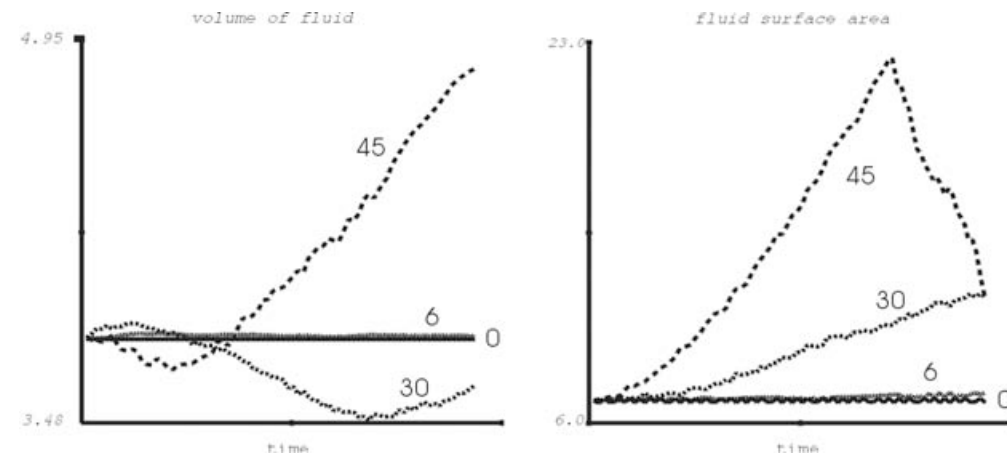


Figure 3. Total fluid volume (left) and interface area evolution for the four cases shown in Fig. 2.

The evaluation of volume fluxes in a given direction by the standard method in *FLOW-3D*[®] is at least as high as that attributed to the PLIC (piecewise linear interface calculation) method and it also conserves fluid volume. However, the accuracy deteriorates when fluid is moved at an angle to a coordinate direction. Figure 2 shows the shape of a round 2D droplet after moving it at different angles to the horizontal axis. The droplet is moved in a uniform velocity field a distance equal to about five times its initial diameter. The mesh consists of square cells with the size equal to 1/10 of the droplet diameter. It is clear that moving along the axis (zero angle) produces minimal distortion, which proves the accuracy of the flux calculation based on a given distribution of fluid fraction function.

The worst case occurs at 45 degree angle. The reason for the distortion of the droplet shape is the use of the split algorithm and of old F values in the calculation of fluxes in each direction. Figure 3 shows the total fluid volume and surface area as a function of time for the four angles. The

volume decreases because cells overflow and the extra fluid volume is discarded, while an increase in volume results from over-emptying cells. These errors are of the first-order with respect to the time step size

Surface area changes with time can be used as a measure of the distortion on the original droplet.

3. Semi-Lagrangian advection method

The new VOF advection method consists of three steps:

1. Approximate fluid interface in a cell with a planar surface;
2. Move the fluid volume according to the local velocity field;
3. Compute new fluid fraction values in the computational cells using an overlay procedure.

The interface in a donor cell is approximated with a planar surface

$$n_x x + n_y y + n_z z = C \quad (4)$$

where $\mathbf{n}=(n_x, n_y, n_z)$ is the unit outward normal to the interface, x , y and z are coordinates of points located on the surface and C is a constant. The normal can be computed as follows:

$$\begin{aligned} n_x &= \frac{1}{|\mathbf{n}|} \frac{F_r - F_l}{\Delta x} \\ F_l &= \frac{2(1 - F_i)(1 - F_{i-1})}{2 - F_i - F_{i-1}} \\ F_r &= \frac{2(1 - F_i)(1 - F_{i+1})}{2 - F_i - F_{i+1}} \end{aligned} \quad (5)$$

In Eq. (5) Δx is the cell size in the x -direction, F_l and F_r represent portions of the left and right cell faces not occupied by fluid, and i is the cell index for the x -direction (Fig. 4a). For example, if either cell i or $i-1$ is full (*i.e.* F_i or F_{i-1} are equal to 1.0), then F_l is equal to zero. Fluid fraction values from the previous time step, $t=t_n$, are used in Eq. (5).

Similar computations are done for the y and z components of the normal.

The value of C is found iteratively to match the volume of fluid in the cell bounded by the plane to the actual amount of fluid in cell.

The next step is to move the fluid volume in the cell. This is done by moving each face of the cell using the velocity component located at the face, as shown in Fig. 4b. For example, the left face is moved using the U_{i-1} velocity component, and the right face is moved using the U_i component.

The distance dx , by which a cell face is moved, is computed using a second order integration of the equation

$$\frac{dx}{dt} = \frac{A_x}{V_f} U \quad (6)$$

For the right cell face of cell i the distance is computed as

$$dx = \frac{A_{xi}}{V_{fa}} \frac{U_i \Delta t}{1 - \frac{1}{2} \frac{\partial U}{\partial x} \Delta t} \quad (7)$$

$$\frac{\partial U}{\partial x} = \frac{U_i - U_{i-1}}{\Delta x}$$

The expression for dx is derived by using a Taylor expansion for the U velocity with respect to dx , retaining only zero- and first-order terms, and using the average velocity along the distance dx to integrate Eq. (6). The result is second-order accurate with respect to cell size Δx .

Volume fraction V_{fa} is taken from the cell *upstream* of the cell face.

The velocity component U_i is assumed constant along the face so the cell face is moved without changing its orientation, *i.e.* it remains parallel to itself. Therefore, the volume of the cell after the translation (dashed outline in Fig. 4c) may, in general, be different than the original cell volume, even if the velocity divergence in the cell is equal to zero:

$$dV = V_{old} / V_{new} \neq 1.0 \quad (8)$$

Due to the change of the aspect ratio of the advected control volume, the normal \mathbf{n} to the interface may change its direction, as shown in Fig. 4c. This is taken into account when solving Eqs. (4) and (5) for C thus allowing for fluid compression and stretching during advection (fluid fraction values in Eq. (5) are still taken from the solution at the previous time step).

The final step is to apportion the advected fluid volume to the new, acceptor cells. This is done with the help of Eq. (4). The fact that the orientations of the cell faces do not change during advection greatly simplifies the task. An adjustment of the computed fluid volumes is made using dV computed in Eq. (8) to make sure that the combined volume of fluid in the acceptor cells is equal to the volume in the donor cell.

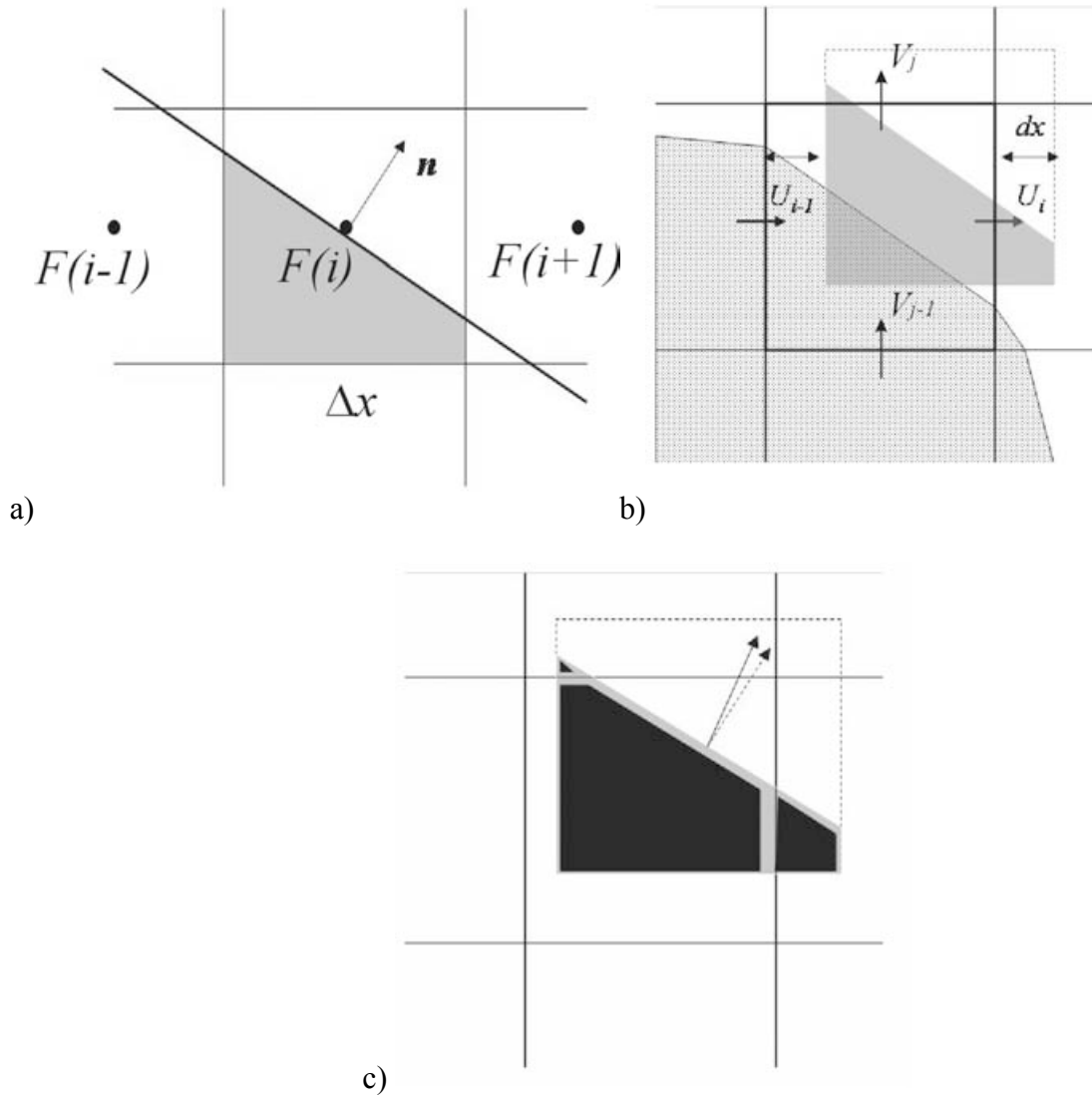


Figure 4. Three steps of the semi-Lagrangian interface tracking method: a) piecewise linear interface reconstruction with the normal \mathbf{n} ; b) moving the control volume, and c) overlaying the advected volume onto the grid.

If a donor cell does not contain an interface, then the algorithm is reduced to a three-dimensional first-order donor advection where fluid is assumed to be evenly distributed across the cell volume. In this case, the first step described above in this section (interface reconstruction) is not required. The third step is simplified in that the volume of fluid apportioned to a given acceptor cell is equal to the size of the portion of the advected volume overlapping the acceptor cell times the fluid fraction in the donor cell. The second step does not change.

Just like the standard method, the new algorithm conserves fluid volume in the local sense, that is, when moving fluid between donor and acceptor cells – fluid volume leaving a donor cell is equal to the volume placed into the acceptor cells.

Each cell containing fluid is used as a donor only once during the procedure automatically ensuring that a donor cell cannot be over-emptied. However, the same cell may be an acceptor for more than one donor cell, creating a possibility for overfilling. Advected volumes from different donor cells may overlap each other placing their fluid into the same physical space. If the total fluid volume in an acceptor cell exceeds the cell volume, it is discarded and the excess volume is added to the volume error. As with the standard method, the cumulative volume error decreases the time step size.

The reason for advected fluid volumes overlapping comes from moving cell faces parallel to themselves. Therefore, swirling flows away from an interface (cells are full already) are areas where such errors can be expected to be the largest. This is in contrast to the standard advection method, where most volume errors originate near the interface.

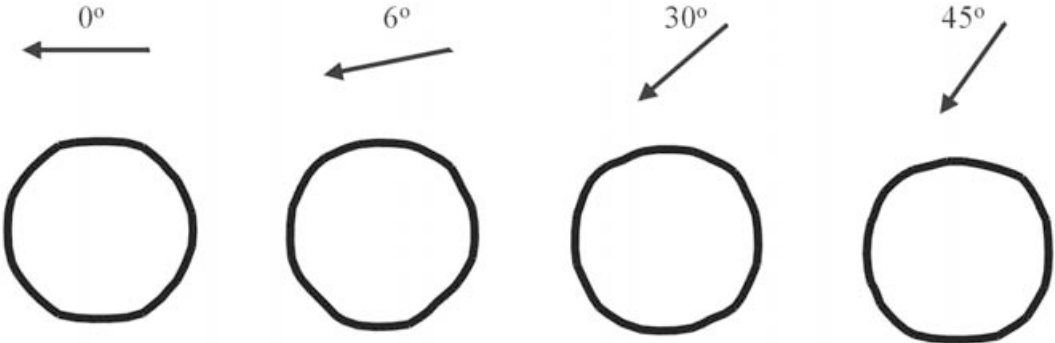


Figure 5. Same as in Fig.2 using the new VOF method.

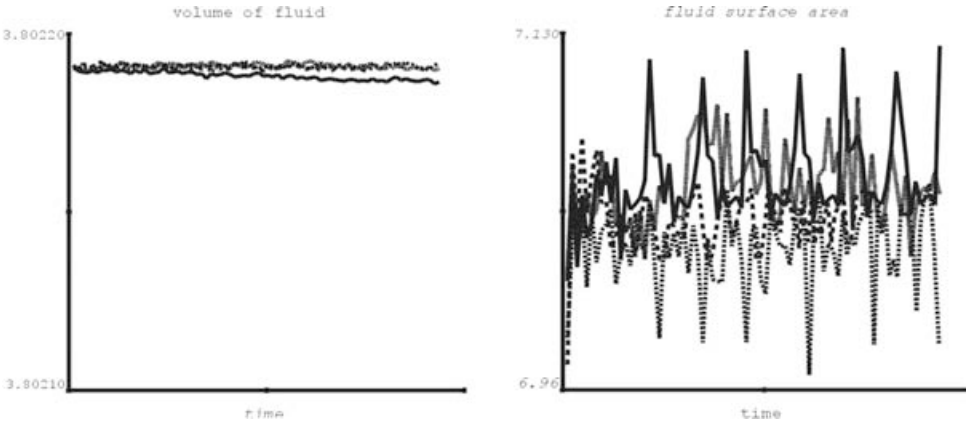


Figure 6. Total fluid volume (left) and interface area evolution for the four cases shown in Fig. 5.

4. Examples

The first test is to run the same problem of the round droplet advection shown in Figs. 2 and 3. The results are shown in Figs. 5 and 6, respectively. The geometry of the droplet is well preserved in all cases. The volume and area curves for different flow directions are so close that it is hard to tell one line from another, at least on the scale used for plotting. The main point is that the variation in volume and area for all four angles is small – within 0.0002 % for the volume and within 1 % for the area, which is the same level of error for the zero-degree case using the standard method.

In another 2D test a round droplet starts moving diagonally through the grid and then impinges on a rectangular obstacle. Figure 7 shows the initial and final configuration of the fluid for the standard (left) and the semi-Lagrangian methods. The volume error for the two cases is compared in Fig. 8.

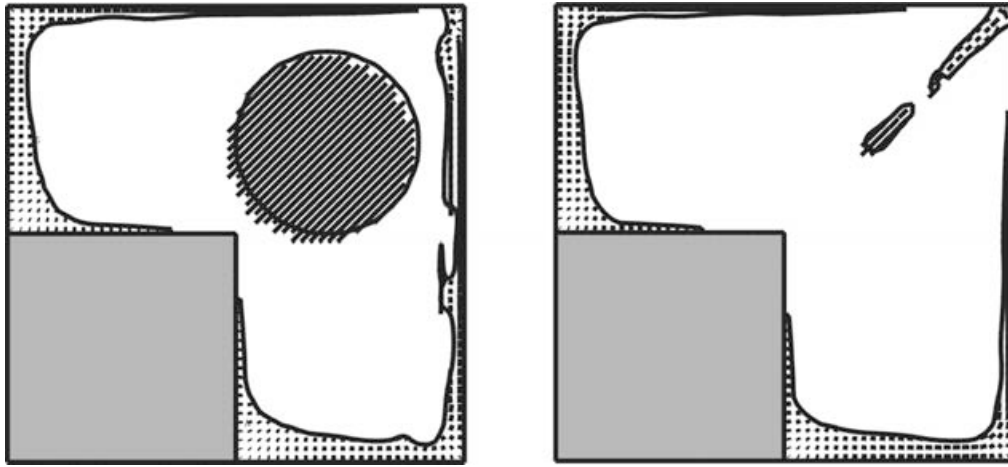


Figure 7. Fluid interface shape after moving a round droplet against a square obstacle using the standard (left) and the semi-Lagrangian method (right). The initial droplet position and velocity has been added to the frame on the left.

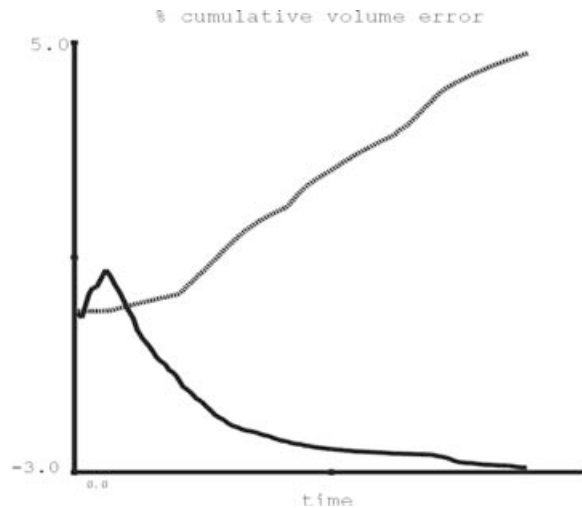


Figure 8. Total fluid volume as a function of time for the two cases shown in Fig. 7. The solid line represents the standard method.

The new advection method predicts a more symmetric and smooth interface shape. As expected, the volume error is positive at all times. Note that the cumulative volume error for the old method is the sum of both negative and positive errors in individual cells, which partially cancel each other when integrated over the mesh and time.

The final test presented here is a 3D flow of a jet emerging from a round hole in a flat plate. The mesh is uniform with the cell size equal to 0.4 of the jet diameter. Two cases were considered: with and without surface tension. The shapes of the jet, after it moved a distance equal to about 25 times its diameter, are shown in Fig. 9.



Figure 9. The shape of a fluid jet emerging from a round hole after it moves a distance equal to 25 lengths of its diameter. Top left - standard method, top right – semi-Lagrangian method, bottom left – standard method with surface tension, bottom right – standard method with surface tension.

The standard method without surface tension (top left) shows a more or less coherent jet, with small spurious fluctuations developing on its surface that at this point start to generate small satellite droplets moving alongside the jet. Even though these perturbations are small compared to the jet size, their location and magnitude appear to be random and asymmetric. They appear about 8 to 10 jet diameters downstream from the plate. Prior to this location the jet is round and smooth.

The jet shape predicted by the semi-Lagrangian method (top right) is also smooth and round to about 10 diameters downstream from the hole. After that point very regular and symmetric perturbations develop on the surface, with their magnitude growing linearly with distance along the jet. These perturbations have the shape of small bumps, a cell or two in size, that protrude

normal to the jet axis, at 45 degree angles to the coordinate directions. They are attributed to insufficient accuracy in the computation of the surface normal in the relatively coarse grid.

The remaining two pictures show the jet shape in the presence of surface tension. In the standard method (bottom left) the surface tension manages to preserve the shape of the jet for longer than without it – 18-20 diameters. However, eventually interface distortion due to advection feeds into the surface tension model and results in a rapid breakup of the jet in the remaining 5-7 diameters of its length.

The in the new method the jet distortion still starts at about 8 diameters in length, but grows slower along its remaining length and the final breakup is somewhat less dramatic than in the standard method.

In all tests above the CPU time used by the new method is within +/-3 % of that used by the standard method

5. Conclusions

A new unsplit interface tracking method based a Lagrangian interface reconstruction and three-dimensional advection has been developed and incorporated into **FLOW-3D**[®] version 8.2. The new model is activated by setting IFVOF=5 in the input file. Its main advantage over the standard method (IFVOF=3 or 4) is an increased accuracy in moving fluid interfaces in arbitrary directions with the rectangular grid.

One of the simplifications in the new methods, that fluid volume in each cell is advected as a rectangular shape results in positive volume errors due to cell overfilling. Most of the volume error is generated in full cells, away from fluid interface, in flow areas with high vorticity. This is in contrast to the standard method, where most of the volume errors appear in the vicinity of the interface.

An accurate surface normal evaluation is one of the cornerstones of interface tracking methods and even more so when the piecewise linear interface reconstruction is used. 3D tests indicate that surface normal calculation in the new algorithm may need further improvement, especially in cases when fluid is resolved by only a couple of cells.

The new method is not currently used as a default for any model in **FLOW-3D**[®]. It is compatible with all other options, *e.g.*, turbulence, thermal energy and scalar models. However, these quantities are always advected using first-order spatial approximations (in other words, using IFENRG=3 will produce the same results as IFENRG=2). It can be also used for both sharp-interface and no-sharp-interface cases (ITB=1 and 0), and for one- and two- fluid cases.

References

- [1] C.W. Hirt and B.D. Nichols, “*Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries*,” J. Comp. Phys., **39**, 201-225,1981.

- [2] Ph. Colella, “*Multidimensional Upwind Methods for Hyperbolic Conservation Laws*,” J. Comp. Phys., **87**, 171-200, 1990.
- [3] E.G. Puckett, A.S. Almgren, J.B. Bell, D.L. Marcus, W.J. Rider, “*A High-Order Projection Method for Tracking Fluid Interfaces in Variable Density Incompressible Flows*,” J. Comp. Phys., **130**, 269-282.
- [4] J.E. Pilliod, E.G. Puckett, “*Second-Order Accurate Volume-of-Fluid Algorithms for Tracking Material Interfaces*,” submitted to J. Comp. Phys., 1998.