INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2001; **36**: 373–389 (DOI: 10.1002/fld.32)

A level set approach for computing solutions to inviscid compressible flow with moving solid boundary using fixed Cartesian grids

Meng-Hsuan Chung*,[1]

National Center for High-Performance Computing, PO Box 19-136, Hsinchu, Taiwan, Republic of China

SUMMARY

A level set approach for computing solutions to inviscid compressible flow with moving solid surface is presented. The solid surface is considered to be sharp and is described as the zero level set of a smooth explicit function of space and time. The finite volume TVD–MacCormack's two-step procedure is used. The boundary conditions on the solid surface are easily implemented by defining the smooth level set function. The present treatment of the level set method allows the handling of fluid flows in the presence of irregularly shaped solid boundaries, escaping from the bookkeeping complexity in the so-called 'surface-tracking' method. Using the proposed numerical techniques, a two-dimensional numerical simulation is made to investigate the aerodynamic phenomena induced by two high-speed trains passing by each other in a tunnel. Copyright © 2001 John Wiley & Sons, Ltd.

KEY WORDS: fixed Cartesian grids; inviscid flow; level set approach; moving solid boundary

1. INTRODUCTION

A numerical method is developed for computing the motion of inviscid compressible flow embedded with complex-shaped solid surfaces. Instead of explicitly tracking the solid boundary on a fixed Cartesian grid, the author intends to implicitly 'capture' the boundary using a level set approach. The boundary will be identified as the zero level set of a smooth function.

Over the years, the challenges posed by fluid flow problems involving moving solid boundaries have resulted in some computational techniques directed toward their solution. Numerical methods applied to such problems are required to follow the arbitrary motion of the complex boundary shapes. A chimera grid embedding technique has been proposed by Benek [1,2], where a boundary-fitted structured grid is constructed closely around the solid

* Correspondence to: National Space Program Office, Hsinchu, Taiwan, Republic of China.

[1] E-mail: mhsuan@ms19.hinet.net

Received September 1998
Revised May 1999

Copyright © 2001 John Wiley & Sons, Ltd.

boundary and is overlapped with an outer structured grid. The inner grid always follows the solid-body motion and communicates with the fixed outer grid in the overlapped region. Such a boundary conforming grid arrangement is convenient and accurate; the solid boundary is always maintained and boundary conditions are applied at the exact locations without any smearing or redistribution. The chimera method has been employed in a wide variety of applications: wing–body–tail configurations [2], general viscous flows [3], weapon separation off of a wing [4], flow field around F-18 aircraft [5], flow field of space shuttle orbiter vertical tail [6], numerical simulation of a tilting disk heart valve [7], and environmental flow problems [8]. The method has been adapted for solutions of Reynolds-averaged Navier–Stokes equations for three-dimensional incompressible flows by Chen *et al.* [9].

The methods based on single fixed Cartesian grids have also been found in some literature. One major limitation of these methods is that complex flow geometries are difficult to incorporate. The representation of a solid boundary on a fixed Cartesian grid layout calls for special treatment, since the control volumes through which the irregular boundary passes become fragmented in solid and liquid regions. Quirk [10] detailed some of the procedures involved in dealing with cell fragments or cut cells in the framework of compressible inviscid flows around stationary obstacles. However, the calculation of fluxes is not presented in detail in that work. Also, in connection with compressible inviscid flows, Pember *et al.* [11] used local refinement and a redistribution procedure to enforce conservation in the vicinity of the solid–fluid boundary. Arbitrary-shaped boundaries moving through a Cartesian mesh have also been dealt with using finite element methods by Young *et al.* [12]. In Zeeuw and Powell [13] and Bayyuk *et al.* [14], a Cartesian grid is employed to track the motion of solid objects through an inviscid compressible fluid. Among others, a distinct branch has been started since the works of Viecelli [15,16] in which a shear-free surface is modeled in a marker and cell technique. The pressure along some desired boundary is used to enforce the no-through flow condition; if fluid flows through the boundary, the pressure is increased on the boundary until the through flow ceases. Along this line, Goldstein *et al.* [17,18] have employed a modified immersed boundary technique (Peskin [19,20]) to impose no-slip boundary conditions over arbitrary-shaped solid boundaries via a feedback external concentrated force field. Turbulent flows around solid obstacles have been calculated using this approach and the results have been shown to compare favorably with experiments. However, the gains of feedback control must be chosen carefully and manually in an unsteady flow problem. The method, as used in these works, treats the solid boundary as a series of steps conforming with the Cartesian grid layout, i.e., in piecewise constant fashion. In a finite volume setting, Udaykumar and Shyy [21,22] used a cut-cell technique to compute incompressible viscous flows in arbitrary geometries. In their works, it was demonstrated that by means of a suitably defined consistent mosaic of control volumes, it is possible to obtain accurate solutions for flows on Cartesian meshes.

All of the aforementioned methods cannot avoid the problem of bookkeeping for capturing the moving solid boundary. Recently, a new set of algorithms for following propagating interfaces has been developed. A Hamilton–Jacobi level set formulation for moving interfaces was introduced (Osher and Sethian [23]). These algorithms handle topological merging and breaking naturally, work in any number of space dimensions, and do not require that the moving front be written as a function. Instead, the moving front is embedded as a particular

Copyright © 2001 John Wiley & Sons, Ltd.

level set in a fixed domain partial differential equation. Numerical techniques borrowed from hyperbolic conservation laws are then used to accurately calculate the correct solution that satisfies the global entropy condition for propagating fronts given in Sethian [24]. These schemes have been used to model a variety of problems in front motion [23], flame propagation [24], the geometry of moving surfaces [25], detonation shock [26], and vortex motion modeling [27]. Following the introduction of this level set formulation for moving fronts, it has also been used for theoretical analysis of motion by mean curvature [28,29] and for constructing minimal surfaces [30]. For an overview of level set methods, refer to References [31,32].

In this paper, the author adopts the notion of level set function to implicitly capture the moving solid boundary and impose the required boundary conditions for computing two-dimensional inviscid compressible flows. The idea is proved amenable by a self-justifying example, i.e., impulsively starting subsonic flow over a circular cylinder. Then it is applied to simulate a transient flow field induced by two high-speed trains passing by each other in a tunnel.

2. GOVERNING EQUATIONS

Considering the unit length of the solid boundary, far-field density ρ_∞, far-field velocity u_∞, and $\rho_\infty u_\infty^2$ being the reference quantities for length, density, velocity, and pressure respectively, the two-dimensional non-dimensional integral form of the conservation law for an inviscid compressible flow can be written as

$$\frac{\partial}{\partial t} \oiiint \mathbf{U}\,\mathrm{d}V + \oiint_S \Phi\cdot\mathrm{d}\mathbf{S} = 0 \tag{1}$$

where V is the domain considered and S is the boundary of V. The conservative variable vector $\mathbf{U}$ and the flux vectors $\mathbf{F}$ and $\mathbf{G}$ in the flux $\Phi = \mathbf{F}\vec{\imath} + \mathbf{G}\vec{\jmath}$ are defined as

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho uu + p \\ \rho uv \\ \rho uH \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \rho v \\ \rho vu + p \\ \rho vv \\ \rho vH \end{bmatrix} \tag{2}$$

where $E = e + [(u^2 + v^2)/2]$ is the total energy per unit mass, e is the internal energy per unit mass, and $H = h + [(u^2 + v^2)/2] = E + (p/\rho)$ is the total enthalpy per unit mass. The thermodynamic equation of state of a perfect gas, $p = (\gamma - 1)\rho e$ with $\gamma = 1.402$, is used to close these conservative equations. $\vec{\imath}$ ($\vec{\jmath}$) are the unit vectors directed toward the positive x (y) co-ordinate axis.

The boundary conditions for the computational domain are demonstrated as follows:

1. At a solid wall boundary, the relative normal velocity is zero. Hence, only one eigenvalue of the system of Euler equations is positive and only one physical condition can be

Copyright © 2001 John Wiley & Sons, Ltd.

imposed, namely $(\mathbf{u}-\mathbf{v})\cdot\mathbf{n}=\mathbf{0}$, where $\mathbf{u}$ is the velocity of fluid and $\mathbf{v}$ and $\mathbf{n}$ are the velocity vector and the unit normal vector of the solid boundary respectively. The other variables at the wall, in particular velocity and pressure, have to be determined by extrapolation from the interior to the boundary. In this work, because the focus of interest is to develop a whole numerical procedure, a zero-order extrapolation along the normal direction is used for simplicity. It is equivalent to $\partial p/\partial n=0$ and $\partial v_{\mathrm{t}}/\partial n=0$, where v_{t} denotes the relative tangential velocity. Then, for density, $\partial \rho/\partial n=0$ from the assumption of adiabatic walls.

2. On the far-field boundary, the one-dimensional Riemann invariants method is locally applied to ensure the non-reflection of unphysical waves [33].

3. NUMERICAL METHOD

3.1. TVD–MacCormack scheme

In a finite volume setting, Equation (1) is solved by the MacCormack's two-step procedure [34], which reads, for the predictor of the forward–forward version

predictor:

$$\mathbf{U}^{p}_{i,j}=\mathbf{U}^{n}_{i,j}-\lambda(\mathbf{F}^{n}_{i+1,j}\cdot\mathbf{S}_{i+1/2}+\mathbf{F}^{n}_{i,j}\cdot\mathbf{S}_{i-1/2}+\mathbf{G}^{n}_{i,j+1}\cdot\mathbf{S}_{j+1/2}+\mathbf{G}^{n}_{i,j}\cdot\mathbf{S}_{j-1/2} \tag{3}$$

corrector:

$$\mathbf{U}^{c}_{i,j}=\mathbf{U}^{n}_{i,j}-\lambda(\mathbf{F}^{p}_{i,j}\cdot\mathbf{S}_{i+1/2}+\mathbf{F}^{p}_{i-1,j}\cdot\mathbf{S}_{i-1/2}+\mathbf{G}^{p}_{i,j}\cdot\mathbf{S}_{j+1/2}+\mathbf{G}^{p}_{i,j-1}\cdot\mathbf{S}_{j-1/2} \tag{4}$$

average:

$$\mathbf{U}^{n+1}_{i,j}=\frac{1}{2}(\mathbf{U}^{p}_{i,j}+\mathbf{U}^{c}_{i,j}) \tag{5}$$

where $\lambda=\Delta t/h^{2}$ and h is the mesh size of the finite volume cell. The implementation, in practice, is to alternate forward/backward difference for the predictor in two co-ordinate directions, thus four combinations are completed in four time steps. Further, the order of combination is reversed in the next four time steps to make the solution as smooth as possible. Finally, eight time steps constitute one cycle of combination.

To render the solution oscillation-free and retain second-order accuracy in space and time almost everywhere (except at extrema points), a total variation diminishing (TVD) dissipation flux with Roe-average cell-face variable and 'minmod' limiter is added into the scheme [35].

As to the boundary conditions, they are easily implemented with the aid of one more layer of phantom cells at the far-field and fixed-wall boundaries. The remaining problem is how to apply the wall boundary conditions at the irregularly shaped moving solid boundary on a Cartesian grid layout. The approach is demonstrated and discussed in the next subsections.

3.2. *Level set function*

Before proceeding to its applications in this work, the theories and characteristics related to the level set function are introduced first. Consider a boundary, either a curve in two dimensions or a surface in three dimensions, separating one region from another, and imagine that this curve/surface moves in its normal direction with a known speed function F. The goal is to track the motion of this interface as it evolves. The motion of the interface in its normal direction is the only concern, and the tangential motion will be ignored. This speed F can depend on the *local* properties of the interface, such as curvature and normal direction, or the *global* properties of the interface, such as integrals along it and associated differential equations, or the *independent* properties, such as an underlying fluid velocity, which passively transports the interface.

Now the level set formulation is derived. Given an initial position for an interface Γ, where Γ is a closed curve in R^2, and a speed function F, which gives the speed of Γ in its normal direction, the level set method takes the perspective of viewing Γ as the zero level set of a function $\phi(\mathbf{r}, t=0)$ from R^2 to R. That is, let $\phi(\mathbf{r}, t=0) = \pm d$, where d is the distance from $\mathbf{r}$ to Γ, and the plus (minus) sign is chosen if the point $\mathbf{r}$ is outside (inside) the initial hypersurface Γ. Then, by the chain rule, an evolution equation for the interface may be produced, namely

$$\phi_t + F|\nabla\phi| = 0 \tag{6}$$

$$\phi(\mathbf{r}, t=0) = \text{given} \tag{7}$$

This is an initial value partial differential equation in one higher dimension than the original problem. There are several advantages to this level set perspective:

1. A discrete grid can be used together with a finite volume to devise a numerical scheme to approximate the solution.
2. Intrinsic geometric properties of the interface are easily determined from the level set function ϕ. For example, the normal vector is given by $\mathbf{n} = \nabla\phi/|\nabla\phi|$.
3. The formulation is unchanged for propagating interfaces in three dimensions.

Turning to the work in this paper, the solid boundary is regarded as the interface separating the solid region and the fluid one. Then, the speed function F depends on the normal direction and is defined as

$$F = \mathbf{v}\cdot\mathbf{n} \tag{8}$$

This definition can be extended off the solid boundary by defining the normal vector $\mathbf{N}$ as the local one at a point $\mathbf{r}$ in the domain determined by the particular level set passing through that point and constructing the velocity field $\mathbf{V}$ satisfying the rigid-body motion of the object, that is

$$\mathbf{N} = \nabla\phi/|\nabla\phi| \tag{9}$$

Copyright © 2001 John Wiley & Sons, Ltd.

$$\mathbf{V} = \mathbf{V}_c + \Omega \times \mathbf{r}' \tag{10}$$

where $\mathbf{V}_c$ denotes the solid-body translational velocity, Ω the solid-body rotational angular velocity, and $\mathbf{r}'$ the displacement of $\mathbf{x}$ from the center of solid-body rotation. Thus, the domain of definition has been enlarged and

$$F = \mathbf{V}\cdot\mathbf{N} \tag{11}$$

Meanwhile, Equations (6) and (7) can now be satisfied by the solution

$$\phi(\mathbf{r}, t) = \phi_0(\mathbf{r}^*(t)) \tag{12}$$

where

$$\phi_0(\mathbf{r}) \equiv \phi(\mathbf{r}, 0) \tag{13}$$

$$\mathbf{r}^*(t) = \Theta^{-1}(t)[\mathbf{r} - \mathbf{r}_c(t)] \tag{14}$$

$$\Theta(t) = \begin{bmatrix} \cos\theta(t) & -\sin\theta(t) \\ \sin\theta(t) & \cos\theta(t) \end{bmatrix} \tag{15}$$

$$\mathbf{r}_c(t) = \int_0^t \mathbf{V}_c(\tau)\,\mathrm{d}\tau \tag{16}$$

$$\theta(t) = \int_0^t \Omega(\tau)\,\mathrm{d}\tau \tag{17}$$

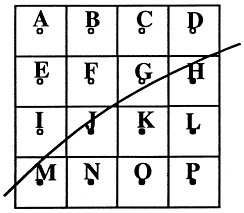
Note that $\phi_0(\mathbf{r})$ needs not to be a distance function as usual because the information of distance has no use in the present approach as seen in the next subsection. Nevertheless, the important characteristics (9) still holds.

Equations (12)–(17) mean that the evolution of the level set function is fully governed by the rigid-body motion, i.e., unlike the case of free surface flows, the speed function F is independent of $\mathbf{u}$ and p. This property makes it possible to trace back the level set field at any instant to the initial level set field along the trajectory of rigid-body motion (see Equation (12)). If ϕ_0 is an analytical function, so is ϕ and one can save lot of computational time from solving the evolutionary equation (6). If ϕ_0 is given only at discrete points, then it must be determined by interpolation.

3.3. Boundary conditions on the solid–fluid interface

By exploiting the level set function, an approach has been proposed to implement the boundary conditions on the solid–fluid interface. At first, the finite volume cells are sorted into four categories according to the value of the level set function at the node itself and its immediate four neighbors (Figure 1):

Copyright © 2001 John Wiley & Sons, Ltd.



● (solid)	$\phi < 0$
○ (fluid)	$\phi > 0$
Outer cell	A, B, C, E
Outer surface cell	D, F, G, I
Inner cell	L, N, O, P
Inner surface cell	H, J, K, M

Figure 1. Four types of cells classified by the value of ϕ at the node and its four immediate neighbors.

(i) internal cell if $\phi_{i,j} \leq 0$ and ($\phi_{i\pm1,j} \leq 0$ and $\phi_{i,j\pm1} \leq 0$)
(ii) internal surface cell if $\phi_{i,j} \leq 0$ and ($\phi_{i-1,j} > 0$ or $\phi_{i+1,j} > 0$ or $\phi_{i,j-1} > 0$ or $\phi_{i,j+1} > 0$)
(iii) outer surface cell if $\phi_{i,j} > 0$ and ($\phi_{i-1,j} \leq 0$ or $\phi_{i+1,j} \leq 0$ or $\phi_{i,j-1} \leq 0$ or $\phi_{i,j+1} \leq 0$)
(iv) outer cell if $\phi_{i,j} > 0$ and ($\phi_{i\pm1,j} > 0$ and $\phi_{i,j\pm1} > 0$)

In the fluid region ($\phi_{i,j} > 0$), there is a problem in updating the flow variables for the outer cells which are one or two mesh sizes away from the internal surface cells. That is, the flow variables are needed at these internal surface cells to close the algebraic equation system that resulted from applying the flow solver mentioned above. The strategy of determining these flow variables is stated as follows (see Figure 2):

1. For each internal surface cell centered at C, calculate $\nabla\phi$ using the central difference scheme, then the unit normal vector $\mathbf{n} = \nabla\phi/|\nabla\phi|$. Meanwhile, the shape of the solid boundary crossing point C is classified and point 1 is defined according to Figure 2.

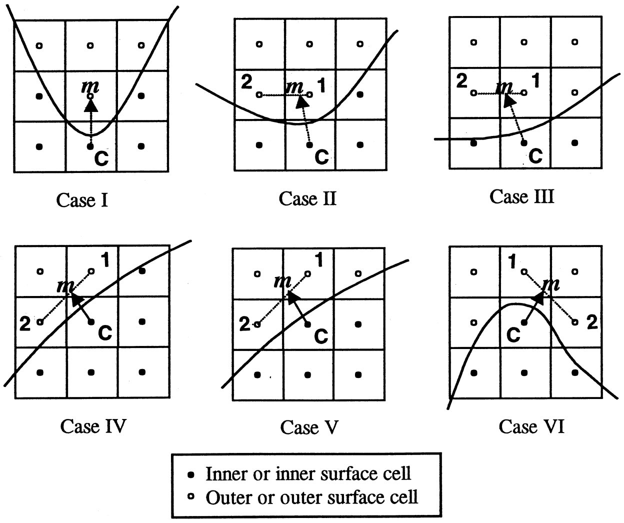


Figure 2. Various cases of solid boundary crossing through the mesh cells.

2. Point 2, an outer cell that has the largest ϕ except point 1, is then determined. The implication of such a choice is that the normal vector will cross through between points 1 and 2.
3. Then point m is found. For Case I, it is just the center of the neighboring outer surface cell; for the other cases, it is the intersection of the normal line with the line connecting points 1 and 2.
4. The flow variables (p, ρ, and tangential velocity) at the internal surface cell C are set as those at point m. Except for Case I, they are linearly interpolated from points 1 and 2. If f represents any flow variable, then

$$f_m = f_1 + r(f_2 - f_1) \tag{18}$$

where

$$r = \begin{cases} |\phi_x/\phi_y|_\mathrm{c} & \text{if } y_1 = y_2 \\ |\phi_y/\phi_x|_\mathrm{c} & \text{if } x_1 = x_2 \end{cases} \quad \text{for Cases II and III} \tag{19}$$

$$r = \begin{cases} 1/(1 + |\phi_y/\phi_x|_\mathrm{c}) & \text{if } x_1 = x_\mathrm{c} \\ 1/(1 + |\phi_x/\phi_y|_\mathrm{c}) & \text{if } y_1 = y_\mathrm{c} \end{cases} \quad \text{for Cases IV, V, and VI} \tag{20}$$

5. The normal velocity at C is obtained from the no-through flow relation $(\mathbf{u} - \mathbf{v}) \cdot \mathbf{n} = 0$.

Note that the information at the internal cells is not needed and the TVD dissipation in some co-ordinate direction is included for the outer surface cell computation only if all the neighboring cell information in that direction is available.

3.4. Summary

The algorithm can now be summarized.

Step 1. Initialize a smooth function $\phi(\mathbf{r}, 0)$, which has a zero value on the solid boundary.
Step 2. Set up the boundary conditions on the far-field, fixed-wall, and the embedded solid–fluid interface.
Step 3. Update the conservative flow variables using the TVD–MacCormack scheme.
Step 4. Construct a new smooth function $\phi^{(n+1)}$ by Equations (12)–(17).
Step 5. Now one time step is advanced. The zero level set of $\phi^{(n+1)}$ gives the new solid-boundary position.
Step 6. Repeat Steps 2–5 until the required time.

Because the solid boundary crosses through the meshes irregularly, the upper bound of the time step for numerical stability is much less than that obtained from the conventional

Copyright © 2001 John Wiley & Sons, Ltd.

Courant–Friedrich–Lewy (CFL) condition. For the time being, the time step is chosen manually by trial and error.

4. NUMERICAL RESULTS

4.1. Impulsively starting subsonic flow over a circular cylinder

This is a case used to test the applicability of the present level set approach. The far upstream Mach number is 0.405 and the reference length for non-dimensionalization is the radius of the cylinder. Firstly, as a base for comparison with the fixed Cartesian grid approach, the O-type body-fitted grid is used to solve this problem. The instantaneous non-dimensional pressure contour at non-dimensional time $t = 1$ is shown in Figure 3. Further, the effect of grid refinement is investigated by comparing the pressure distribution on the surface of the circular cylinder for each grid (Figure 4). It can be concluded that the convergent solution has been almost reached for the finest grid resolution (96 radial grids × 128 circumferential grids). In all of the above calculations, $\Delta t = 5 \times 10^{-4}$.

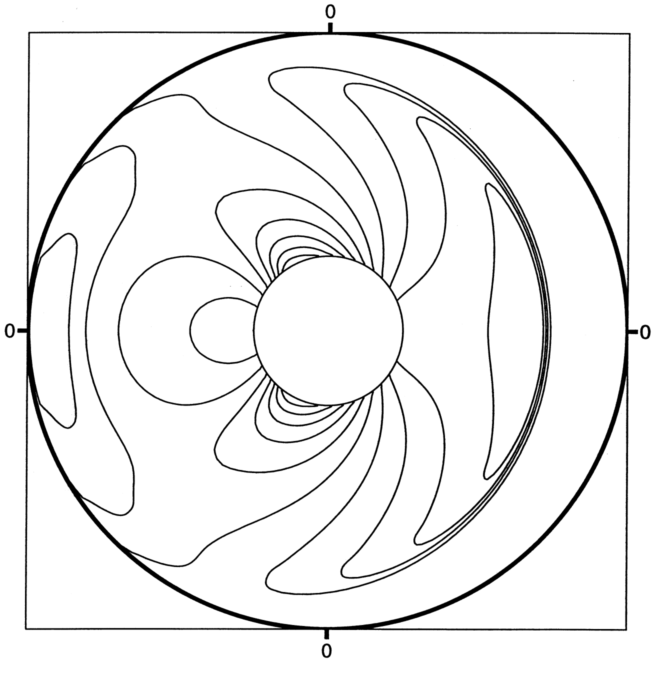


Figure 3. Instantaneous pressure contour at $t = 1$ for flow over a circular cylinder using the body-fitted grid approach (grid resolution = 96 × 128).

Copyright © 2001 John Wiley & Sons, Ltd.

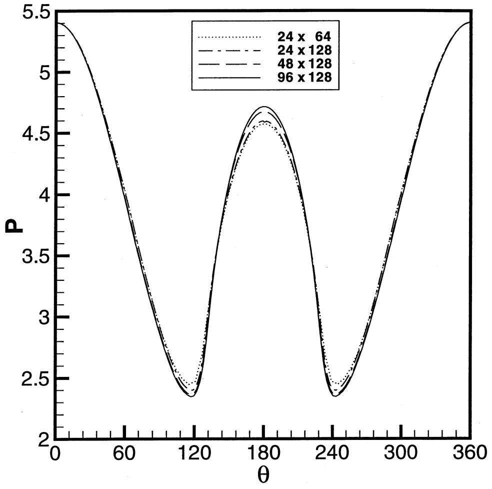


Figure 4. Pressure distribution on the surface of the circular cylinder at $t = 1$ using the body-fitted grid approach. Also shown is the grid refinement effect.

Secondly, the level set approach using fixed uniform Cartesian grids is used to solve the same physical problem. The cylinder moves with $\mathbf{V}_c = \mathbf{i}$ and $\Omega = \mathbf{0}$ in a quiet ambience, and from Equations (12)–(17), its moving boundary can be represented by the zero level set of a smooth function

$$\phi(x, y, t) = \sqrt{(x - t)^2 + y^2} - 1 \tag{21}$$

The Δt used in these calculations are 5×10^{-4} for 64×64 mesh, 2×10^{-4} for 128×128 mesh, and 1×10^{-4} ($t < 0.1$) then 5×10^{-4} for 192×192 and 256×256 meshes. Similarly, the instantaneous pressure contour at $t = 1$ is shown in Figure 5. The effect of grid refinement is shown in Figure 6. It can be said that the mesh with 256×256 cells will meet the requirement of convergence well enough. The cragginess in these curves reflects the inherent nature pertaining to this Cartesian grid approach. During the motion, in general, a finite volume cell will experience different categories defined in Figure 1, and the case of solid boundary crossing through the mesh cells would change correspondingly (Figure 2). This case switching causes the change of interpolated point-pair, point 1 and 2, hence the small oscillation observed in Figure 6.

The two approaches are compared in Figure 7 for their finest grids. They fit in with each other except the roughness of mesh size scale. In practice, almost the whole solution domain has not been polluted by this grid-scale oscillation, as can be seen in Figure 5. As another rigorous check, the unsteady pressure at four locations on the surface is plotted in Figure 8 for these two approaches. It is seen that, immediately after the starting peak, they agree very well except at $\theta = \pm 90°$. The large pressure gradient near these sites, as seen in Figure 5, makes the

Copyright © 2001 John Wiley & Sons, Ltd.

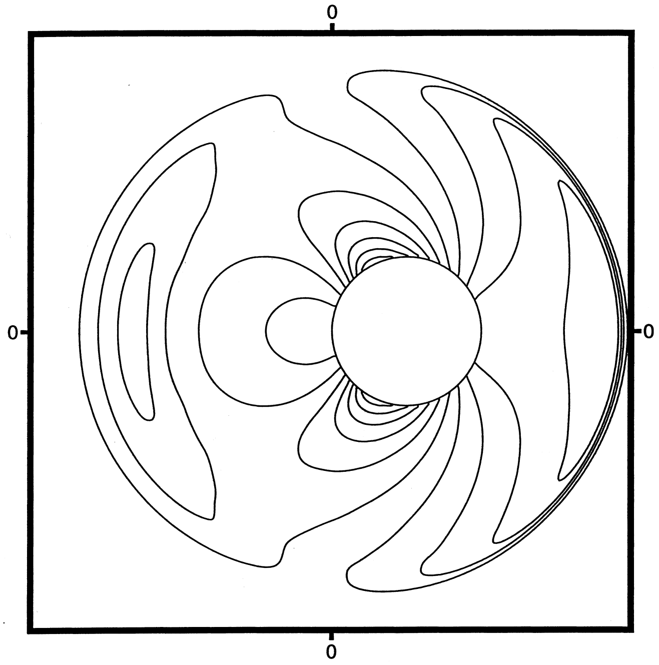


Figure 5. Instantaneous pressure contour at $t = 1$ for flow over a circular cylinder using the fixed Cartesian grid level set approach (grid resolution = 256 × 256).

interpolation error more severe, hence explains the slight deviation between the two approaches.

4.2. Two trains passing by each other in a tunnel

As a preliminary application, the flow field induced by two high speed trains passing by each other in a tunnel is calculated using the present level set approach. The problem definition and the computational domain are illustrated in Figure 9. Two trains have the same shape and move at the same speed of Mach number 0.283 (350 km h^{-1}). In this work, the half-length and the half-width of the train body, L and W respectively, are set at 12 and 1.5 m respectively and a semi-circle is substituted for the actual head/tail profile of the train.

The flow field where the same trains pass by each other at the same speed is point symmetric about the center point, i.e.

$$\begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}_{\mathbf{r}'} = \begin{bmatrix} \rho \\ -\rho u \\ -\rho v \\ \rho E \end{bmatrix}_{\mathbf{r}} \tag{22}$$

Copyright © 2001 John Wiley & Sons, Ltd.

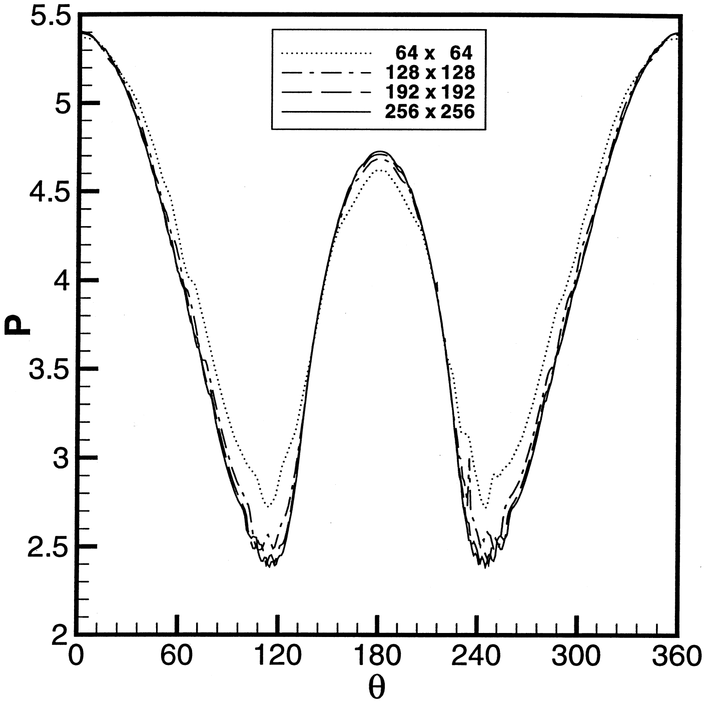


Figure 6. Pressure distribution on the surface of the circular cylinder at $t = 1$ using the fixed Cartesian grid level set approach. Also shown is the grid refinement effect.

where $\mathbf{r}'$ is the point of point symmetry of $\mathbf{r}$ as shown in Figure 9. Only the velocities u and v are anti-symmetric. The computation is done with this boundary condition. Only the negative y flow field is solved and both the grid points and the CPU time are reduced.

To reduce the compressive pressure wave induced by impulsive starting, the trains are located far away from each other at the beginning and moves at the velocity of $\mathbf{V}_c = 0.02t\mathbf{i}$ for $0 < t \leq 50$ and $\mathbf{V}_c = \mathbf{i}$ for $50 < t$. In this work, the center of the train body (x_c, y_c) is initially $(-55, -2.25)$.

The train surface boundary can be represented by the zero level set of a smooth function, i.e.

$$\phi(x, y, t) = \begin{cases} \sqrt{[x - (x_c \pm L)]^2 + (y - y_c)^2} - W & \text{if } \pm(x - x_c) > L \\ \pm(y - y_c) - W & \text{if } |x - x_c| \leq L \quad \text{and} \quad \pm(y - y_c) \geq 0 \end{cases} \tag{23}$$

where $x_c = -55 + 0.01t^2$ for $0 < t \leq 50$, $x_c = t - 80$ for $50 < t$, and $y_c = -2.25$. A total of 2000×80 cells is used and $\Delta t = 0.01$ for most of the computational stage and a smaller value during the time immediately after the beginning.

Copyright © 2001 John Wiley & Sons, Ltd.

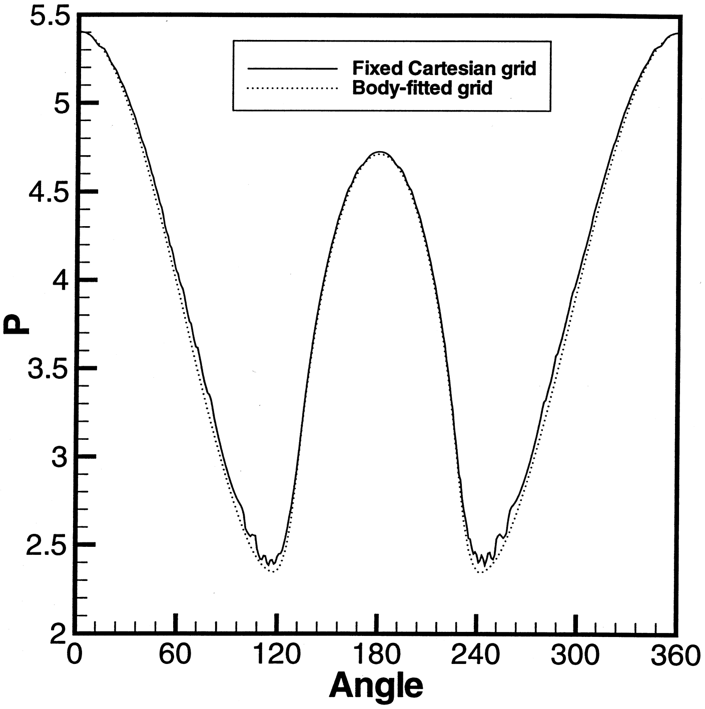


Figure 7. Comparison between the fixed Cartesian grid and the body-fitted grid approach.

Figure 10 shows the pressure contour plots at a series of consecutive instants. It can be seen that very complex unsteady flow fields occur in this problem. Four locations on the train surface are chosen for quantitative investigation as shown in Figure 9. The history of pressure at these locations is recorded and plotted versus the distance between two front noses of train (Figure 11). Near the train noses (point A and B), the surface boundary curve is more or less perpendicular to the velocity of motion, thus causes frequent case switching aforementioned in the circular cylinder example. At the outer side (point C) or the meeting side (point D), there is no such switching during the whole computational period. In Figure 11, this numerical aspect is clearly illustrated by the corresponding small-oscillation or smooth curves. The three vertical dotted lines denote the times when the front noses meet each other, the front nose meets the rear nose of the other train, and the rear noses meet each other respectively. Except for the outer side location, obviously, there is a period of low pressure while the recording location comes to face to the train on the other track. The response at the outer side location is very complicated and the recovery to a steady condition seems to need more time than the other locations.

Copyright © 2001 John Wiley & Sons, Ltd.

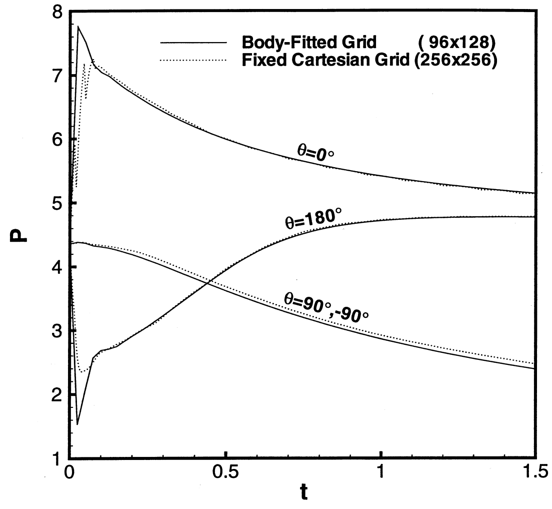


Figure 8. Comparison of pressure history at four locations on the cylinder surface between the body-fitted grid and the fixed Cartesian grid approach.

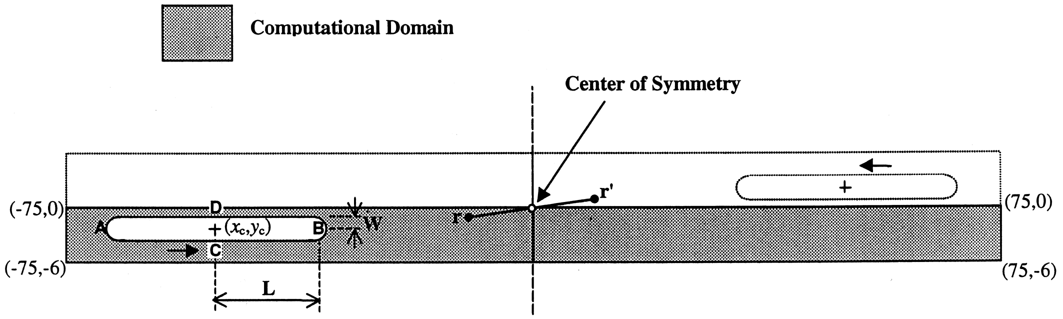
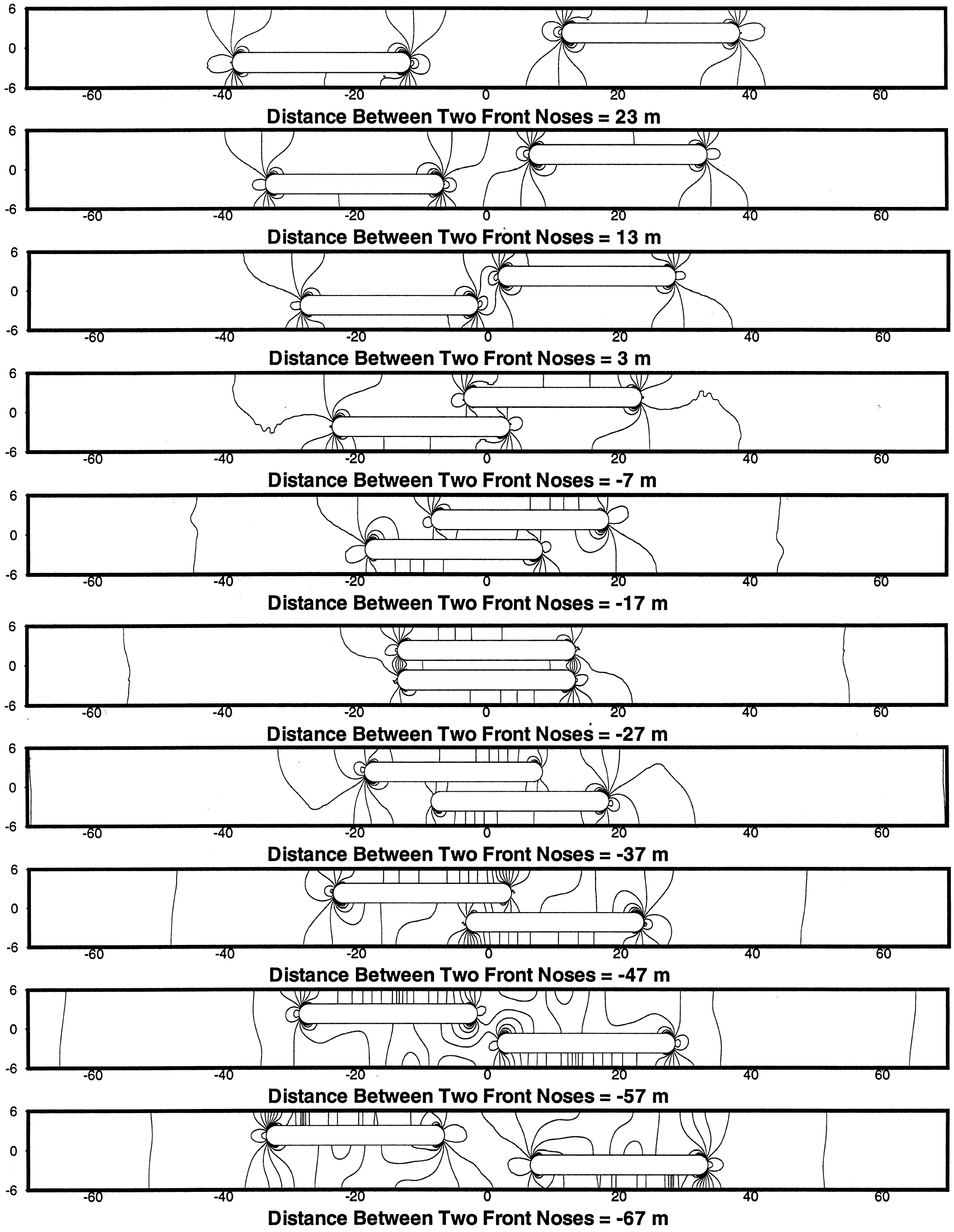


Figure 9. Problem definition and computational domain for two trains passing by each other in a tunnel. A, B, C, and D denote the four locations to record the pressure history.

5. DISCUSSIONS AND CONCLUSIONS

In this paper, the solutions to two-dimensional inviscid compressible flow with moving solid boundary is calculated using a novel level set approach where the level set function needs not to be a distance function, but is limited to an explicit function of spatial position to avoid solving the evolutionary equation. In real applications, this limitation could be of little significance if a general solid boundary can be approximated by some superposition of analytical geometric shapes.

Copyright © 2001 John Wiley & Sons, Ltd.



Distance Between Two Front Noses = 23 m

Distance Between Two Front Noses = 13 m

Distance Between Two Front Noses = 3 m

Distance Between Two Front Noses = -7 m

Distance Between Two Front Noses = -17 m

Distance Between Two Front Noses = -27 m

Distance Between Two Front Noses = -37 m

Distance Between Two Front Noses = -47 m

Distance Between Two Front Noses = -57 m

Distance Between Two Front Noses = -67 m

Figure 10. Time sequence of pressure contour for two trains passing by each other in a tunnel.

Copyright © 2001 John Wiley & Sons, Ltd.

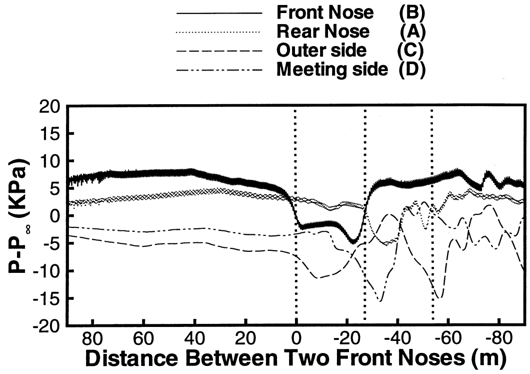


Figure 11. Pressure history at four locations on the train body surface. Location A, B, C, and D are defined in Figure 9.

The idea is proved applicable by the test case of a subsonic flow over a circular cylinder. The simulation of a transient flow field induced by two high-speed trains passing by each other in a tunnel illustrates that the present method can capture the complex unsteady phenomenon. In principle, the notion behind the present approach can be applied in a general non-orthogonal grid layout. That work is under way along with the extension to three-dimensional space.

REFERENCES

1. Benek JA, Steger L, Doughertj FC, Buning PG. Chimera: a grid-embedding technique. Arnold Engineering Development Center Report, AEDC-TR-85-64, Arnolds Air Force Station, TN, 1985.
2. Benek JA, Buning PG, Steger JL. A chimera grid embedding technique. AIAA Paper No 85-1523, 1985.
3. Benek JA, Donegan TL, Suhs NE. Extended chimera grid embedding scheme with application to viscous flows. AIAA Paper No 87-1126, 1987.
4. Dougherty FC, Benek JA, Steger JL. On application of chimera grid schemes to store separation. NASA Report, TM-88193, 1985.
5. Cummings RM, Rizk YM, Schiff LB, Chaderjian NM. Navier–Stokes predictions of the flowfield around the F-18 (HARV) wing and fuselage at large incidence. AIAA Paper No 90-0099, 1990.
6. Parks SJ, Buning PG, Steger JL, Chan WM. Collar grids for intersecting geometric components within the chimera overlapped grid scheme. AIAA Paper No 91-1587, 1991.
7. Kiris C, Chang ID, Rogers SE, Kwak D. Numerical simulation of the incompressible internal flow through a tilting disk valve. AIAA Paper No 90-0682, 1990.
8. Meakin RL, Street RL. Simulation of environmental flow problems in geometrically complex domains. Part 2: a domain splitting method. *Methods in Applied Mechanics and Engineering* 1988; **68**: 311–331.
9. Chen HC, Patel VC, Ju S. Solutions of Reynolds-averaged Navier–Stokes equations for three-dimensional incompressible flows. *Journal of Computational Physics* 1990; **88**(2): 305–336.
10. Quirk JJ. An alternative to unstructured grids for computing gas dynamic flows around arbitrarily complex two-dimensional bodies. ICASE Report No. 92-7, NASA Langley Research Center, Hampton, VA, 1992.
11. Pember RB, Bell JB, ColeIla P. An adaptive Cartesian grid method for unsteady compressible flow in irregular regions. *Journal of Computational Physics* 1995; **120**: 278–304.
12. Young DP, Melvin RG, Bieterman MB, Johnson FT, Samant SS, Bussoletti JE. A locally refined rectangular grid finite element method: application to computational fluid dynamics and computational physics. *Journal of Computational Physics* 1992; **92**: 1.

Copyright © 2001 John Wiley & Sons, Ltd.

13. Zeeuw DD, Powell KG. An adaptively-refined Cartesian mesh solver for the Euler equations. AIAA Paper No 90-0000, 1990.
14. Bayyuk A, Powell KG, van Leer B. A simulation technique for 2D unsteady inviscid flows around arbitrarily moving and deforming bodies of arbitrary geometry. AIAA Paper No 93-3391, 1993.
15. Goldstein D, Handler R, Sirovich L. Modeling a no-slip surface with an external force field. *Journal of Computational Physics* 1993; **105**: 354–366.
16. Goldstein D, Handler R, Sirovich F. Direct numerical simulation of turbulent flow over a modeled riblet covered surface. *Journal of Fluid Mechanics* 1995; **302**: 333.
17. Viecelli JA. A method for including arbitrary external boundaries in the MAC incompressible fluid computing technique. *Journal of Computational Physics* 1969; **4**: 543–551.
18. Viecelli JA. A computing method for incompressible flows bounded by moving walls. *Journal of Computational Physics* 1971; **8**: 119–143.
19. Peskin CS. Flow patterns around heart valves: a numerical method. *Journal of Computational Physics* 1972; **10**: 252–271.
20. Peskin CS. Numerical analysis of blood flow in the heart. *Journal of Computational Physics* 1977; **25**: 220–252.
21. Udaykumar HS, Shyy W. A grid-supported marker particle scheme for interface tracking. *Numerical Heat Transfer B* 1995; **27**(2): 127.
22. Udaykumar HS, Kan H-C, Shyy W, Tran-Son-Tay R. Multiphase dynamics in arbitrary geometries on fixed Cartesian grids. *Journal of Computational Physics* 1997; **137**: 366–405.
23. Osher S, Sethian JA. Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulation. *Journal of Computational Physics* 1988; **79**(1): 12–49.
24. Sethian JA. Curvature and the evolution of fronts. *Communications in Mathematics and Physics* 1985; **101**: 487–499.
25. Sethian JA. Numerical algorithms for propagating interfaces: Hamilton–Jacobi equations and conservation laws. *Journal of Differential Geometry* 1989; **31**: 131–161.
26. Chen VG, Giga V, Goto S. Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations. *Journal of Differential Geometry* 1991; **33**: 749.
27. Fink PT, Sob WK. *Proceedings of the Royal Society of London A* 1978; **362**: 195.
28. Sethian JA. Computing the motion of curves and evolving surfaces. In *Geometric Motion*, Almgren FA, Taylor J (eds). American Mathematical Society: Providence, RI, 1991.
29. Sethian JA, Adalsteinsson D. An overview of level set methods for etching, deposition, and lithography development. *IEEE Transactions on Semiconductor Manufacturing* 1997; **10**(1): 167–184.
30. Aslam TD, Bdzil JB, Stewart DS. Level set methods applied to modeling detonation shock. *Journal of Computational Physics* 1996; **126**(2): 390.
31. Sethian JA. Level set methods. *Journal of Fluid Mechanics* 1997; **345**: 413.
32. Harabetian E, Osher S, Shu CW. An Eulerian approach for vortex motion using a level set regularization procedure. *Journal of Computational Physics* 1996; **127**(1): 15.
33. Hirsch C. *Numerical Computation of Internal and External Flows*, *Vol. 2: Computational Methods for Inviscid and Viscous Flows*. Wiley: New York, 1992.
34. Rizzi, Inouye. Time split finite volume method for three-dimensional blunt-body flows. *AIAA Journal* 1973; **11**: 1478–1485.
35. Garcia-Navarro P, Alcrudo F, Saviron JM. 1D open-channel flow simulating using TVD–MacCormack scheme. *Journal of Hydraulic Engineering* 1992; **118**(10): 1359–1372.

Copyright © 2001 John Wiley & Sons, Ltd.