

Numerical method for complex moving boundary problems in a Cartesian fixed grid

Kensuke Yokoi

Computer and Information Division, RIKEN (The Institute of Physical and Chemical Research), Wako 351-0198, Japan

(Received 9 November 2001; published 22 May 2002; publisher error corrected 3 June 2002)

We propose a numerical method to capture and track complex moving boundaries advected in flows. Our method is based on the level set method, but it overcomes the problem of accumulation of reinitialization error in the level set method.

DOI: 10.1103/PhysRevE.65.055701

PACS number(s): 46.15.-x, 02.70.Bf, 83.85.Pt, 95.75.Pq

Fluid phenomena with complex moving boundaries (gas-liquid, liquid-liquid, gas-liquid-solid, etc.) appear everywhere, such as flows in rivers, kitchens, bodies, and plants. Studies on these phenomena are considered challenging and are important for various fields such as physics, industry, and medicine. However, numerical studies on these phenomena are difficult because two different scales of a phenomena and a complex moving boundary must be treated simultaneously, and boundary conditions of complex shapes must be considered. Therefore, numerical studies for these problems have hardly been done. In this work, a numerical algorithm for these problems is proposed.

In our approach, a Cartesian fixed grid is used. One advantage of using such a grid is that a grid for complex moving boundaries need not be generated. Large deformation and topological change of interfaces can be treated because the grid is never distorted. However, there are some problems such as numerical diffusion at the interface and influence of the grid. Various techniques have been proposed to overcome such problems. In these techniques, we consider the level set method [1,2] to be suitable for complex moving boundary problems. The level set method has been applied to various moving boundary problems [2–4].

The level set method is an interface-capturing method. The method expresses the surface of an $(N-1)$ dimension as the zero contour of an N -dimensional function. The function and the zero contour are called the level set function and the zero-level set, respectively. In the level set method, the interface (zero-level set) is not tracked directly but implicitly through the level set function. In this paper, the function that satisfies Eq. (1) is used as a level set function f_{ls}

$$\begin{aligned} f_{ls} &= 0 \quad \text{at the interface,} \\ |\nabla f_{ls}| &= 1 \quad \text{for the whole region.} \end{aligned} \quad (1)$$

The function is also called a signed distance function that represents the distance from the interface. For example, Fig. 3(d) shows the level set function of a cut-out cylinder (initial condition of the Zalesak problem) as shown in Figs. 3(a)–3(c). The time evolution of the level set function is calculated by the advection equation

$$\frac{\partial f_{ls}}{\partial t} + \mathbf{u} \cdot \nabla f_{ls} = 0. \quad (2)$$

The level set method has various features. The method can automatically treat topological change in interfaces without special treatment. To precisely capture interfaces in a Cartesian grid, smoothing of the interface must be introduced. In the level set formulation, the thickness of the interface can be easily maintained because the distance from the interface can be obtained from the distance function. The level set method is useful for estimating the curvature κ . To compute the curvature, the divergence of a normal vector to the interface is required. In the formulation, the normal vector is always well defined from the level set function. The curvature is calculated as follows:

$$\kappa = \nabla \cdot \left(\frac{\nabla f_{ls}}{|\nabla f_{ls}|} \right). \quad (3)$$

From the curvature, the surface tension force can be calculated easily by using the continuum surface force model [5].

Although the level set method is a useful interface-capturing method as mentioned above, there is a problem of reinitialization error, which is discussed below. The initial level set function f_{ls} satisfies the property of the signed distance function. However, after calculation of the advection equation (2), f_{ls} does generally not satisfy the feature of the signed distance function as shown in Fig. 1(b). Therefore, the function must be rebuilt as the signed distance function without changing the position of the zero-level set; here, it is assumed that the zero-level set has represented the interface. This procedure is called reinitialization [3]. However, the position of the zero-level set moves slightly in this procedure [Fig. 1(c)]. This error is the reinitialization error. Although

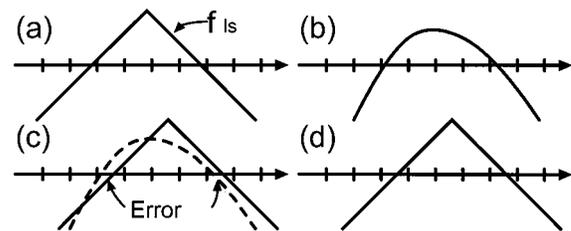


FIG. 1. Schematic figure of reinitialization by the original level set method.

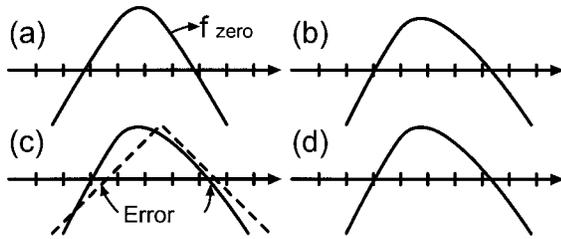


FIG. 2. Schematic figure of the present method.

the error is small, the error accumulates in each time step because the next time evolution is calculated from f_{ls} with the error [Fig. 1(d)]. Therefore, the accumulation of errors is not negligible in some cases [for example, Figs. 5 and 6(c)]. The accumulation of errors will be a serious problem in cases of complex moving boundaries.

In this Rapid Communication, we propose a method to prevent the accumulation of reinitialization errors. First, an outline of the numerical method is given. After the explanation of the numerical procedure of the method, a comparison of results obtained using the present method and those obtained using the original method is presented.

An outline of our method is presented using Fig. 2. In our method, a function f_{zero} that is not reinitialized is used to track the interface. The interface is tracked through the zero-contour of f_{zero} [13]. The time evolution of f_{zero} is calculated by the advection equation [(a)→(b) in Fig. 2]. To capture the interface (i.e., to compute curvature, smoothing etc.), the level set function f_{ls} is constructed from the zero contour of f_{zero} [Fig. 2(c)]. This gives rise to an error similar to the error that occurs in the original level set method. However, there is no accumulation of errors in the present method because the time evolution is calculated from f_{zero} without the error [Fig. 2(d)].

We will now explain the numerical procedure. The time evolution of f_{zero} to track the zero-level set is calculated by the advection equation,

$$\frac{\partial f_{zero}}{\partial t} + \mathbf{u} \cdot \nabla f_{zero} = 0, \quad (4)$$

[(a)→(b) in Fig. 2]. As a solver of the advection equation, we use the CIP (cubic interpolated propagation) method [6,7], which is less diffusive and robust. To construct the level set function f_{ls} from the zero contour of f_{zero} [Fig. 2(c)], we use the fast marching method [2,8] and an iterative reinitialization scheme proposed by Sussman *et al.* [9] (referred to hereafter as Sussman's method). Before using these methods, the position of the zero contour of f_{zero} is calculated by interpolating between grids. In this time, linear interpolation is used. From the zero contour, f_{ls} within Δh (where Δh is the grid spacing) from the zero contour of f_{zero} are computed by the fast marching method, solving the Eikonal equation

$$|\nabla f_{ls}| = 1. \quad (5)$$

Other f_{ls} are calculated by Sussman's method. In the calculation, f_{ls} computed by the fast marching method is fixed. Sussman's method solves the following problem to a steady state:

$$\frac{\partial f_{ls}}{\partial \tau} = S(f_{ls})(1 - |\nabla f_{ls}|), \quad (6)$$

where, τ is artificial time and S is a smoothed sign function,

$$S(f_{ls}) = \frac{f_{ls}}{\sqrt{f_{ls}^2 + \epsilon^2}}. \quad (7)$$

In the calculation of Eq. (6), a narrow band approach [2,3] is used. The procedure is repeated using f_{zero} without the reinitialization error as the initial state of the next time step.

We give supplementary explanations of the procedure to construct the level set function f_{ls} from the zero contour of f_{zero} . f_{ls} developed by Eq. (2) should be used as an initial condition of the iterative calculation of Eq. (6) in each time step. If the value of the former time step or $1, -1$ is used as an initial condition of the iteration, the number of iterations will become large. For the calculation of Eq. (2), high-order schemes such as the CIP method may not be needed because the error in the advection calculation disappears in the iterative calculation of Eq. (6). In this paper, we used a first-order upwind scheme. The fast marching method can also be used for the calculation of f_{ls} beyond Δh from the interface. In this time, we did not use this method because it requires more calculation time than that required in the present method. Although the calculation time in Sussman's method is generally short, the reinitialization error is large. However, in our method, the error does not arise because the value

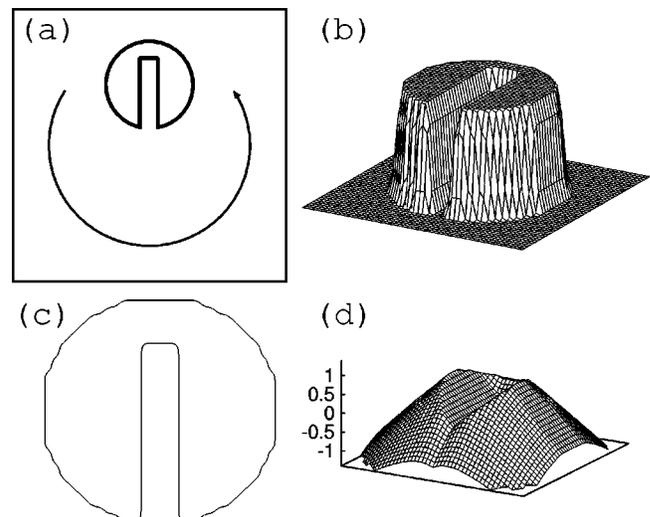


FIG. 3. (a) Configuration of the Zalesak problem. The value of f_{color} inside the cut-out cylinder is 1, others is 0, (b), (c), and (d) are perspective views of f_{color} , the zero-level set and the level set function (signed distance function) of the initial condition, respectively. The rotation speed is such that one complete revolution is performed in 628 steps. Cartesian fixed grids of 100×100 were used.

around the zero-level set computed by the fast marching method has been fixed in the iteration.

A color (density) function f_{color} that is used to define the physical properties for different materials can be generated from a smoothed Heaviside function

$$f_{color} = H_{\alpha}(f_{ls}), \tag{8}$$

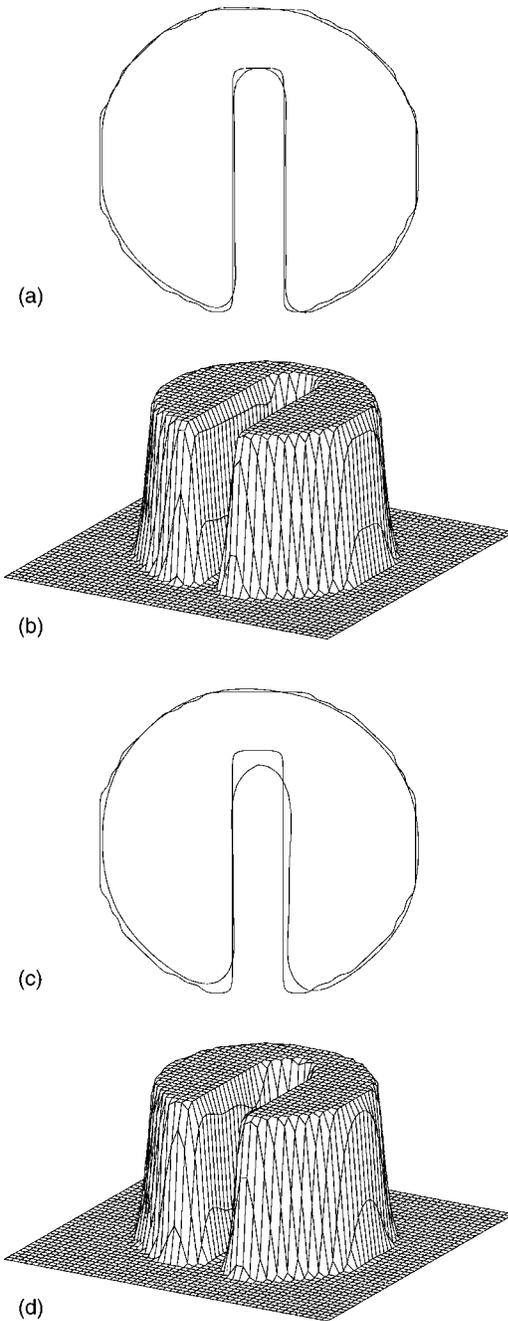


FIG. 4. Results after one revolution [(a) and (b)] and ten revolutions [(c) and (d)] by using the present method. (a) and (c) show the zero-level set with the exact solution. (b) and (d) show the profiles of f_{color} .

$$H_{\alpha}(f_{ls}) = \begin{cases} 0 & \text{if } f_{ls} < -\alpha \\ \frac{1}{2} \left[1 + \frac{f_{ls}}{\alpha} + \frac{1}{\pi} \sin\left(\frac{\pi}{\alpha} f_{ls}\right) \right] & \text{if } |f_{ls}| \leq \alpha \\ 1 & \text{if } f_{ls} > \alpha, \end{cases} \tag{9}$$

where 2α represents the thickness of the interface. In the level set formulation, the thickness can be easily controlled because the level set function has a feature of the distance function.

As a testing problem, we applied the present method to a two-dimensional rigid body rotation problem, called a Zalesak problem [10], as shown in Fig. 3. As an initial condition of the Zalesak problem

$$f_{zero}(x,y) = \begin{cases} 1, & \sqrt{(i-51)^2 + (j-76)^2} < 16 \\ \text{and } j \geq 86 \text{ or } |i-51| > 2 \\ -1, & \text{others,} \end{cases} \tag{10}$$

and $u = 2\pi(j\Delta y - 51\Delta y)$, $v = -2\pi(i\Delta x - 51\Delta x)$ were used. Here u and v are the velocity components of x and y , respectively. Figure 4 shows the results after one revolution and ten revolutions by the present method. The interfaces have been captured well after ten revolutions as well as one revolution.

As a comparison, the original method was also applied to the Zalesak problem. The results are shown in Fig. 5. Deviation of the zero-level set can be seen. The error is the reinitialization error because the same advection equation solver, the CIP method, was used. The error increases with time.

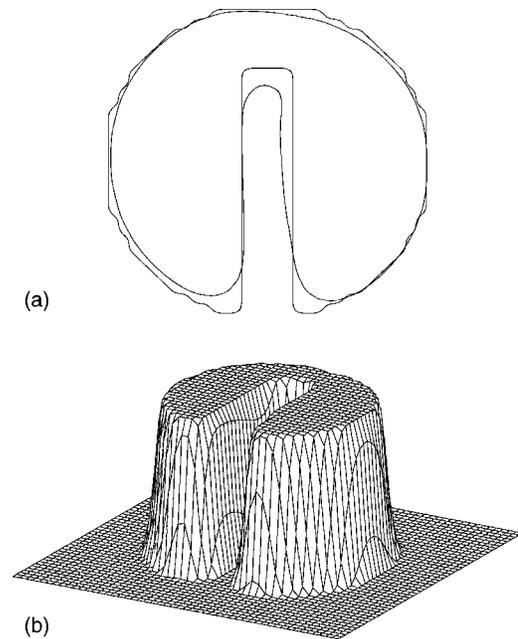


FIG. 5. Zero-level set and f_{color} after one revolution by Sussman's reinitialization with the CIP method.

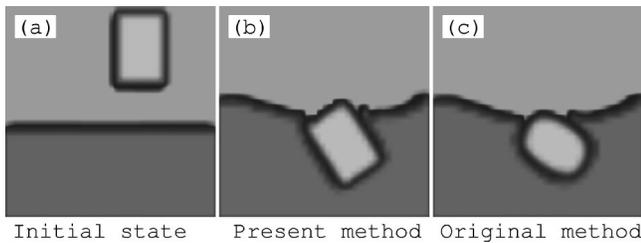


FIG. 6. The rectangular rigid body falling into the liquid, (a) is the initial condition, (b) and (c) are the snapshots after 600 time steps by the present method and the original level set method, respectively. A Cartesian fixed grid of 40×40 was used. The rectangular rigid body is represented by 8×12 meshes. The density ratio of gas:liquid:solid is 1.25:1000:500. The movies are available in Ref. [12].

As another example, the present method was applied to a hydrodynamic problem including a rigid body. In this calculation, we used the CIP-CUP (combined unified procedure) method [11] based on a Poisson equation of the pressure as a

solver of multifluid flows. The results are shown in Fig. 6(b). The present method captured the shape of the rigid body well, though the grid number is small. Figure 6(c) shows the results obtained by using the original method for treatment of the interface of the rigidbody. The results show that the shape cannot be maintained by using the original method because of the reinitialization error.

We have proposed a numerical method based on the level set method for capturing moving interfaces in a Cartesian fixed grid. This method overcomes the problem of accumulation of reinitialization error in the original method. We believe that the method is useful for numerical studies of fluid phenomena with complex moving boundaries, such as bubbling flows, droplet breaking, hydraulic jumps, and blood flows.

I would like to thank Feng Xiao for his useful discussions and comments. Numerical computation for this work was carried out at the Computer and Information Division, RIKEN and the Yukawa Institute for Theoretical Physics, Kyoto University. I am grateful to RIKEN for support.

-
- [1] S. Osher and J.A. Sethian, *J. Comput. Phys.* **79**, 12 (1988).
 [2] J.A. Sethian, *Level Set Methods and Fast Marching Methods* (Cambridge University Press, Cambridge, UK, 1999).
 [3] D.L. Chopp, *J. Comput. Phys.* **106**, 77 (1993).
 [4] K. Yokoi and F. Xiao, *Phys. Rev. E* **61**, R1016 (2000).
 [5] J.U. Brackbill, D.B. Kothe, and C. Zemach, *J. Comput. Phys.* **100**, 335 (1992).
 [6] T. Yabe and T. Aoki, *Comput. Phys. Commun.* **66**, 219 (1991).
 [7] T. Nakamura and T. Yabe, *Comput. Phys. Commun.* **120**, 122 (1999).
 [8] D. Adalsteinsson and J.A. Sethian, *J. Comput. Phys.* **148**, 2 (1999).
 [9] M. Sussman, P. Smereka, and S. Osher, *J. Comput. Phys.* **114**, 146 (1994).
 [10] S.T. Zalesak, *J. Comput. Phys.* **31**, 335 (1979).
 [11] T. Yabe and P.Y. Wang, *J. Phys. Soc. Jpn.* **60**, 2105 (1991).
 [12] <http://hellfire.riken.go.jp/%7Ekensuke> or <http://on.to/kensuke>
 [13] The function f_{zero} and its zero contour are also a level set function and a zero-level set, respectively, in the strict sense. However, here a level set function refers only to a level set function of the signed distance function, and a zero contour refers only to a zero contour of the signed distance function.