Computer Physics Communications 200 (2016) 291–299

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

VOFI — A library to initialize the volume fraction scalar field☆

S. Bnà [a], S. Manservisi [a], R. Scardovelli [a,*], P. Yecko [b], S. Zaleski [c,d]

[a] DIN – Lab. di Montecuccolino, Università di Bologna, Via dei Colli 16, 40136 Bologna, Italy
[b] Physics Department, Cooper Union, New York, NY, USA
[c] Sorbonne Universités, UPMC Univ Paris 06, UMR 7190, Institut Jean Le Rond d'Alembert, F-75005, Paris, France
[d] CNRS, UMR 7190, Institut Jean Le Rond d'Alembert, F-75005, Paris, France

ARTICLE INFO

Article history:
Received 8 July 2014
Received in revised form
27 July 2015
Accepted 28 October 2015
Available online 12 November 2015

Keywords:
Implicit functions
Numerical integration
Volume fraction function
VOF method

ABSTRACT

The VOFI library has been developed to accurately calculate the volume fraction field demarcated by implicitly-defined fluid interfaces in Cartesian grids with cubic cells. The method enlists a number of algorithms to compute the integration limits and the local height function, that is the integrand of a double Gauss–Legendre integration with a variable number of nodes. Tests in two and three dimensions are presented to demonstrate the accuracy of the method and are provided in the software distribution with C/C++ and FORTRAN interfaces.

Program summary

Program title: VOFI

Catalogue identifier: AEYT_v1_0

Program summary URL: http://cpc.cs.qub.ac.uk/summaries/AEYT_v1_0.html

Program obtainable from: CPC Program Library, Queen's University, Belfast, N. Ireland

Licensing provisions: Standard CPC licence, http://cpc.cs.qub.ac.uk/licence/licence.html

No. of lines in distributed program, including test data, etc.: 94963

No. of bytes in distributed program, including test data, etc.: 1679223

Distribution format: tar.gz

Programming language: C, with C++ and FORTRAN interfaces.

Computer: Any computer with a C compiler.

Operating system: Tested on x86 with Linux (openSUSE 13.1, Ubuntu 12.04) and Mac OS X.

Has the code been vectorized or parallelized?: The code does not need any change to be used in parallel with domain decomposition, as done for example in the Paris-Simulator code, http://parissimulator.sf.net, that is massively parallel and uses the Vofi library.

Word size: 64 bits

Classification: 4.11.

Nature of problem: The library computes the volume fraction of a cubic grid cell cut by an interface described by an implicit function.

Solution method: The library computes the integration limits along two coordinate directions and the local height function, that is the integrand of a double Gauss–Legendre integration with a variable number of nodes.

☆ This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (http://www.sciencedirect.com/science/journal/00104655).

* Corresponding author. Tel.: +39 051 2087720; fax: +39 051 2087747.
E-mail address: ruben.scardovelli@unibo.it (R. Scardovelli).

http://dx.doi.org/10.1016/j.cpc.2015.10.026

Restrictions: Cartesian grids with cubic cells.

Running time: Fractions of a second for a grid cell cut by the interface.

1. Introduction

Free-surface and two-phase flows are very common in nature and also in many physics, engineering, chemistry applications. The Volume-of-Fluid (VOF) method is one of several numerical techniques often used to follow the dynamical evolution of interfaces within the framework of direct numerical simulation (DNS) [1,2]. The method is based on the characteristic function $\chi(\mathbf{x}, t)$, a multidimensional Heaviside step function, with value 1 in the reference phase and 0 elsewhere, at a given time t.

The volume fraction function C is the discrete version of χ and it is given by the volume (area in two dimensions) occupied by the reference phase divided by the cell volume, hence $0 \leq C_i(t) \leq 1$, where i is the grid cell index. From the knowledge of the discrete scalar field C it is possible to reconstruct the interface, for example by numerically computing the local outward normal $\mathbf{n}$ as in the Piecewise Linear Interface Calculation (PLIC) method [3], and to estimate the local mean curvature.

The initialization of the volume fraction is trivial in empty cells, $C = 0$, or full cells, $C = 1$, of the computational mesh, while in the cells that are cut by the interface it is necessary to calculate the volume defined by the interface itself and the cell boundary.

A simple, but rather inefficient, initialization of the volume fraction in each cut cell can be achieved by considering an arbitrary number of internal points, either on a regular submesh or randomly distributed. The initial value of C is then given by the ratio of the number of points inside the reference phase with their total number. More advanced techniques for multidimensional integration are based on Monte Carlo methods with different sampling strategies to improve the efficiency of the method [4,5].

An alternative way to initialize the scalar field C in the cut cells is to use recursive local mesh refinement with a simple, linear approximation of the interface at the finest level [6]. However, this approach may require a large number of cell refinements and be very expensive in CPU time if very high accuracy is desired.

Yet another possibility is the direct integration of the analytical expression of the interface. The analytical integration becomes rather involved when the interface is a closed surface, i.e. a bubble or a droplet, which is usually described by an implicit equation. Furthermore, in each cut cell the local limits of integration along one coordinate direction are not independent of the other direction.

In this paper we consider computational grids with cubic cells and present a numerical algorithm to initialize the volume fraction field from a given implicit equation of the interface, $f(\mathbf{x}) = 0$. The algorithm locates the reference phase in the points $\mathbf{x}$ where the function $f(\mathbf{x})$ is negative [7,8].

A first assumption (hypothesis H1) of the Vofi library is that in each cell cut by the interface the implicit equation, $f(\mathbf{x}) = 0$, can be written as $x_1 = g(x_2, x_3)$, where each x_i is one of the three coordinate directions. In these cells, a numerical algorithm determines the main coordinate direction x_1 along which the values of the local height function (defined below) are explicitly computed with a root-finding routine.

The internal and external limits of integration along the other two coordinates, x_2 and x_3, are given by the intersections of the interface with the cell boundary and are computed with standard numerical tools, which include root-finding and minimum search routines.

A second assumption (hypothesis H2) of the Vofi library is that the interface cannot intersect a cell side more than twice. This assumption considerably simplifies the algorithm and it implies that characteristic lengthscales smaller than the grid spacing h_0 are usually not well resolved. At this point it is useful to note that (fact N1) interfaces are reconstructed and advected in a satisfactory way by VOF methods when their local radius of curvature and in general all relevant lengthscales are a few times the value of h_0. Thus, if fact N1 is verified, as expected when the volume fraction field is initialized, then hypotheses H1 and H2 are also verified.

The integration of the local height function determines the volume fraction C in the cell under investigation and it is performed with a double Gauss–Legendre quadrature rule with a variable number of nodes. The two-dimensional problem on a grid with square cells is also included as a special case.

We highlight the fact that even if a VOF/PLIC method can at most reconstruct exactly a linear interface, an accurate initialization of the discrete field C, as provided by the Vofi library, is nevertheless necessary for an exact initialization of the total mass. More importantly, the calculation of geometrical quantities, such as the local interface normal and mean curvature, is often based on finite differences of the volume fraction data, hence very accurate initial data are required to calculate the convergence rate with grid refinement of a given numerical scheme. Finally, we point out that VOF fields are manipulated by a broad range of algorithms. For example, while PLIC method is limited in accuracy, VOF advection schemes are not (see, e.g., [9]).

The library routines are written in `C`, and a simple software interface has been written to make the routines directly available for `FORTRAN` codes. A few two-dimensional and three-dimensional numerical tests are also provided with the software.

2. Numerical method

2.1. Cutoff function value f_h

Let $D \subset R^n$ be a computational domain and Ω the domain occupied by the reference phase defined by

$$\Omega = \{\mathbf{x} \in D : f(\mathbf{x}) \leq 0\}, \tag{1}$$

where the zero level set of $f(\mathbf{x})$ is the interface Γ and level sets with negative values occupy the interior of Ω. The interface is a plane curve when $n = 2$, and a surface when $n = 3$. Three two-dimensional interfaces are shown in Fig. 1, each of them with two level sets corresponding to $f(\mathbf{x}) = \pm l$, with $l = 0.015$. The characteristic function $\chi(\mathbf{x}, t_0)$ at the initial time t_0 is equal to 1 inside Ω and 0 elsewhere.

We then consider a Cartesian subdivision of the domain D with square/cubic cells of side h_0 and area/volume $V_0 = h_0^n$. The volume fraction $C_i(t_0)$ is then defined by the integral

$$C_i(t_0) = \frac{1}{V_0} \int_{V_i} \chi(\mathbf{x}, t_0)\, dV, \tag{2}$$

where V_i is the domain of integration of the ith grid cell. The integration is straightforward in empty or full cells, but in the cell