Computers & Fluids 102 (2014) 182–202

Contents lists available at ScienceDirect

Computers & Fluids

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

A flexible level-set approach for tracking multiple interacting interfaces in embedded boundary methods

Claudia Günther *, Matthias Meinke, Wolfgang Schröder

Institute of Aerodynamics, RWTH Aachen University, Wüllnerstrasse 5a, 52062 Aachen, Germany

ARTICLE INFO

Article history:
Received 28 August 2013
Received in revised form 22 April 2014
Accepted 23 June 2014
Available online 10 July 2014

Keywords:
Level-set method
Complex geometry
Moving boundary
Embedded boundary method
Immersed boundary method
Cut-cell method

ABSTRACT

A versatile multiple level-set method to describe, track, and reconstruct multiple independently moving three-dimensional interfaces in embedded boundary methods for flow simulations is introduced. The use of component based geometry is a key idea of the concept. Multiple level-set functions that are solved separately represent different moving interfaces. A fast assembly procedure provides a combined level-set function which contains all relevant information for the reconstruction of the interfaces on the flow grid. This allows a consistent and accurate treatment of overlapping interfaces and topological changes. A high flexibility and robustness are thus introduced to the method and the preprocessing effort for complex setups is minimized. The accuracy of the intersection information that is reconstructed from the level set representation is shown to be comparable or even superior to a triangulation based reconstruction, depending on the interpolation scheme that is used for the computation of the level-set function on the corner vertices. Regarding the total computing time that is required for the respective interface representation, i.e., including level set transport and reconstruction of the intersection information, the level-set based approach is found to be faster than the triangulation based approach with an appropriately resolved surface triangulation.

1. Introduction

Flows around moving objects or bounded by moving walls and interfaces are relevant in a wide range of scientific and engineering applications. Especially problems which are characterized by large displacements of the boundary and applications which include multiple moving boundaries leading to topological changes of the flow domain are a challenge for computational fluid dynamics (CFD). For such problems, the application of boundary-conforming grid methods almost inevitably leads to considerable errors which result from mesh distortion and interpolation during re-meshing or for overlapping grids. Furthermore, these procedures lead to strongly increased computational costs, which often renders boundary-conforming grid methods infeasible.

Embedded boundary methods, where the geometry is embedded in a non-boundary-conforming fixed background mesh, are an appealing alternative and have gained increasing attention in recent years. Approaches following this key idea are known under many different names, e.g., embedded boundary [1–3], immersed boundary [4–6], immersed/conservative interface [7], cut cell [8–10], ghost cell [11–13], or simply Cartesian (grid) [14–16] methods. Although most of these methods operate on uniform or refined Cartesian grids, there exist a number of approaches for embedded boundaries in arbitrary unstructured meshes [17–19].

While the mesh generation process is substantially simplified for embedded boundary methods, the accurate representation of the embedded boundaries in the background mesh is challenging, since such a representation is not inherently contained in the mesh. The flow solution in the vicinity of the embedded boundaries is addressed in a multitude of individual methods as stated above. At the same time, the description of the interface itself, the tracking of its motion, and the reconstruction of relevant instantaneous information about the interface position on the background mesh is the other essential ingredient for the success of an embedded boundary method [20]. This aspect will be discussed in this paper with a focus on multiple three-dimensional complex moving interfaces embedded in a hierarchically refined Cartesian grid. Unlike Wang et al. [20], we will only consider closed surfaces, i.e., no bodies of infinite thickness. This part of the embedded boundary problem can be treated on a rather general basis, so the method presented in this paper can be applied independently of the approach that is chosen to incorporate the influence of the boundary on the fluid phase. However, the method has been developed in the context of a Cartesian cut cell method

* Corresponding author. Tel.: +49 241 8095410; fax: +49 241 8092257.
E-mail address: office@aia.rwth-aachen.de (C. Günther).

http://dx.doi.org/10.1016/j.compfluid.2014.06.023
0045-7930/

[21,10], and thus some of its features might not be required for other approaches that are less demanding regarding the reconstruction of the boundary on the background mesh.

Since the information about the location of the embedded boundary on the background mesh has to be computed repeatedly at each time step, efficiency is crucial. Furthermore, the preprocessing effort for the user should be minimized. Suitable computational methods strongly depend on the way the interfaces and their motion are described. A basic distinction has to be made between Lagrangian and Eulerian approaches for interface tracking. Many embedded boundary approaches for fluid–structure interaction problems with deforming bodies rely on Lagrangian surfaces to represent the interface, since structural dynamics problems are often formulated in the Lagrangian reference frame, e.g., [3,22–25]. On the other hand, Eulerian interface tracking approaches such as the level-set method are widely used in multiphase and free-surface problems and have recently also been applied to more general embedded boundary problems [26,7,27,28,10].

The description of the geometry using an explicit Lagrangian surface triangulation given in the STereo-Lithography (STL) format seems very attractive and is widely used to simulate embedded complex technical geometries since this allows to directly export the surfaces from the CAD-system [24]. However, this approach has several drawbacks. For example, the computation of the required information about the location of the embedded boundary on the background mesh based on such surfaces is rather expensive. Efficient algorithms for the computation of this information usually have rather high demands on the quality of the surface description. This requires either increased preprocessing efforts to repair incorrect surface representations [29] or more robust and thus expensive reconstruction algorithms [30]. Furthermore, simply advecting the surface mesh points with their respective local velocity can lead to tangling and self-intersection of the mesh [31]. Especially when multiple interfaces move independently and interact with each other, situations will occur that are difficult to deal with in explicit surface representations, such as overlaps of the respective surface meshes and topological changes [32–34]. Finally, although the use of component based geometry, i.e., a complex geometry which is composed of different separate components that are allowed to intersect and perform relative motion, is possible, this comes at the expense of added complexity and costs, since the reconstruction algorithm must be able to extract the relevant part of the geometry, i.e., the wetted surface [35,36].

The application of a level-set method using an implicit surface representation by signed distances resolves some of the above mentioned problems of an explicit Lagrangian surface representation. Due to its inherent ability to cope with topological changes, this approach is very robust. Also, the computation of the interface location on the background mesh can be done very efficiently [7,10]. Furthermore, the availability of the surface distance is very beneficial if a wall distance depending wall model is applied, e.g., for the simulation of turbulent flows using Reynolds-averaged Navier–Stokes (RANS) turbulence modeling approaches [37]. However, the available standard level set approaches are not able to provide a sharp and accurate surface representation if the distance between two individual interfaces is of the order of the mesh resolution, since the surfaces will merge [33]. In this case, solid body representations will be completely corrupted after contact between two bodies or parts has occurred. At the same time, unless special techniques are applied [31], the information to which body a specific part of the modeled interface belongs is usually not available, and the use of component based geometry is impossible since no overlaps of different components can exist.

To overcome the weaknesses of the existing standard level set approaches identified above, we present a novel robust and highly flexible method to describe, track, and reconstruct moving interfaces in embedded boundary methods which uses multiple level-set functions. Our method is targeted on flow problems computed on hierarchically refined Cartesian grids in which multiple three-dimensional complex moving geometries that may interact with each other are embedded. The use of component based geometry is a key idea of the concept. Other than existing embedded boundary approaches, such as [35], the proposed method treats the component based geometry in a simple, robust, and efficient way without the need for complex preprocessing steps. The new method represents different interfaces or components by separate level-set functions, which are solved independently. A special assembly procedure is applied to compute a single combined level-set function which contains all information necessary for the reconstruction of the interfaces on the CFD grid. The concept and ideas underlying the construction and use of this combined level-set function are novel and allow a consistent treatment of the boundaries even if the distance between the interfaces becomes small and topological changes of the flow domain occur. This introduces a greatly increased flexibility and robustness compared to existing methods. At the same time, the necessary preprocessing effort is reduced considerably. Additional information, like the distance between two components of the geometry and the belonging to a specific interface is easily accessible and offers interesting opportunities, such as adaptive refinement of the CFD mesh in narrow gaps. A thorough analysis of the accuracy of the reconstructed intersection information is provided, and an evaluation of the efficiency is conducted.

The remainder of this paper is organized as follows. Section 2 introduces a basic setup for a Cartesian grid embedded boundary flow solver and specifies the necessary inputs, outputs and requirements for an interface tracking and reconstruction algorithm in a general way. Section 3 presents the novel method in detail. Results for the accuracy, efficiency, and robustness of the novel method are presented in Section 4. Finally, the findings of this paper are summarized in Section 5.

2. Embedded boundary flow solver setup for moving interfaces

A general moving boundary flow problem consists of

(I) The governing equations for the fluid flow.
(II) The location/displacement of the interface $\Gamma(t)$ which separates the space occupied by the fluid phase $\Omega^f(t)$ from a different fluid phase or solid domain $\Omega^s(t)$, where t denotes the time.
(III) Suitable boundary conditions to be fulfilled at the interface $\Gamma(t)$ and other domain boundaries.
(IV) Initial conditions for the flow field and the interface location $\Gamma(t_0)$.

For fluid–solid interfaces, (II) can be either prescribed or be a result of fluid–structure interaction. In the latter case, the governing equations for the solid behavior in the domain Ω^s have to be added to the list and (III) and (IV) have to be extended to include suitable boundary and initial conditions for the solid. We will not restrict the problem to a single moving interface $\Gamma(t)$, but consider multiple independently moving interfaces $\Gamma^i(t)$ which are allowed to interact, overlap, and induce topological changes of the fluid domain $\Omega^f(t)$, as sketched in Fig. 1.

Embedded boundary flow solvers discretize and solve this problem in a Eulerian framework on fixed non-boundary conforming meshes which contain, but are not equivalent to, the fluid domain Ω^f, i.e., the meshes are not moving or deforming with respect to the boundary motion. However, the mesh may change due to adaptive