Journal of Computational Physics 231 (2012) 2977–2994

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

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

A conservative coupling algorithm between a compressible flow and a rigid body using an Embedded Boundary method

L. Monasse [a,b,c,*], V. Daru [c,d], C. Mariotti [a], S. Piperno [b], C. Tenaud [c]

[a] CEA DIF, F-91297 Arpajon, France
[b] Université Paris-Est, CERMICS, 77455 Marne-la-Vallée, France
[c] LIMSI – CNRS, 91403 Orsay, France
[d] Arts et Métiers Paris Tech, DynFluid Lab., 75013 Paris, France

ARTICLE INFO

Article history:
Received 23 December 2010
Received in revised form 13 September 2011
Accepted 2 January 2012
Available online 8 January 2012

Keywords:
Compressible flows
Shock capturing scheme
Discrete Element method
Fluid–structure interaction
Embedded Boundary method

ABSTRACT

This paper deals with a new solid–fluid coupling algorithm between a rigid body and an unsteady compressible fluid flow, using an Embedded Boundary method. The coupling with a rigid body is a first step towards the coupling with a Discrete Element method. The flow is computed using a finite volume approach on a Cartesian grid. The expression of numerical fluxes does not affect the general coupling algorithm and we use a one-step high-order scheme proposed by Daru and Tenaud [V. Daru, C. Tenaud, J. Comput. Phys. (2004)]. The Embedded Boundary method is used to integrate the presence of a solid boundary in the fluid. The coupling algorithm is totally explicit and ensures exact mass conservation and a balance of momentum and energy between the fluid and the solid. It is shown that the scheme preserves uniform movement of both fluid and solid and introduces no numerical boundary roughness. The efficiency of the method is demonstrated on challenging one- and two-dimensional benchmarks.

1. Introduction

This work is devoted to the development of a coupling method for fluid–structure interaction in the compressible case. We intend to simulate transient dynamics problems, such as the impact of shock waves onto a structure, with possible fracturing causing the ultimate breaking of the structure. An inviscid fluid flow model is considered, being convenient for treating such short time scale phenomena. The simulation of fluid–structure interaction problems is often computationally challenging due to the generally different numerical methods used for solids and fluids and the instability that may occur when coupling these methods. Monolithic methods have been employed, using an Eulerian formulation for both the solid and the fluid (for instance, the diffusive interface method [16,1]), or a Lagrangian formulation for both the fluid and the solid (for example, the PFEM method [26]), but in general, most solid solvers use Lagrangian formulations and fluid solvers use Eulerian formulations. In this paper we consider the coupling of a Lagrangian solid solver with an Eulerian fluid solver.

For the coupling in space, a possible choice is to deform the fluid domain in order to follow the movement of the solid boundary: the Arbitrary Lagrangian–Eulerian (ALE) method has been developed to that end. It has been widely used for incompressible [11,19] and compressible [15] fluid–structure interaction. However, when solid impact or fracture occur, ALE methods are faced with a change of topology in the fluid domain that requires remeshing and projection of the fluid state

* Corresponding author at: Université Paris-Est, CERMICS, 77455 Marne-la-Vallée, France.
E-mail address: monassel@cermics.enpc.fr (L. Monasse).

0021-9991/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2012.01.002

on the new mesh, which are costly and error prone procedures. Moreover remeshing is poorly adapted to load balancing for parallel computations.

In order to allow for easier fracturing of the solid, we instead choose a method based on fictitious domains that solves the fluid flow on a fixed Eulerian mesh, on which a Lagrangian solid body is superimposed. A special treatment is then applied on fluid cells near the boundary and inside the solid. Different types of fictitious domain methods have been developed over the last 30 years. They can roughly be classified in three main classes: penalization methods, interpolation methods and conservative methods. Among penalization methods, the Immersed Boundary method is certainly the best known and most widely used for fluid–structure interaction. It was originally introduced by Peskin for incompressible blood flows [44,45]. The solid boundaries deform under the action of the fluid velocity, and the presence of the solid adds forces in the fluid formulation that enforce the impermeability of the solid. However, Xu and Wang have pointed out some numerical leaking of fluid into the solid [49]. Following Leveque and Li [33,34], they advocate the use of the Immersed Interface method, which incorporates jump conditions in the finite differences used. However, the absence of fluid mass loss is still not ensured exactly. In a different approach, Olovsson et al. [40,2] couple an Eulerian and a Lagrangian method by penalizing the penetration of the solid into the fluid by a damped spring force. As the stiffness of the spring goes to infinity, the penetration goes to zero. Boiron et al. [5] and Paccou et al. [41] consider the solid as a porous medium, using a Brinkman porosity model. As the porosity goes to zero, the solid becomes impermeable. However, in both cases, as the stiffness grows or the porosity decreases, the use of implicit schemes is mandatory to avoid the severe stability condition of explicit schemes [5,41]. For the high speed phenomena we consider, we use explicit solid and fluid solvers and an explicit coupling algorithm is better suited in order to avoid costly iterative procedures.

A second class of fictitious domain methods consists in enforcing the boundary conditions through interpolations in the vicinity of the boundary, using the exact values taken by the fluid on the boundary [37,13]. The method seems to be very versatile, being used with incompressible Navier–Stokes [37,13], Reynolds-averaged Navier–Stokes [10,29,30], turbulent boundary layer laws [6] and compressible Navier–Stokes [42]. The Ghost Fluid method developed by Fedkiw et al. [18,17] relies on the same type of principle for compressible fluids. The interface is tracked using a level-set function, and conditions are applied on both sides of the interface to interpolate the boundary conditions. The advantage of these methods is that they do not suffer from additional time-step restriction due to stability, and the order of accuracy of the boundary conditions can be set *a priori*. However, the interpolation does not ensure the conservation of mass, momentum and energy in the system. This can cause problems when dealing with shock waves interacting with solids.

In this article, we rather consider the third class of conservative fictitious domain methods, which seems the most adequate framework to develop our coupling algorithm. These methods are generally referred to as Embedded Boundary methods, and they rely on a modified integration of the numerical fluid fluxes in the cells cut by the solid boundary [43,14,22,25]. The original idea of the method can be traced back to Noh's CEL code [39]. The new contribution of the present work consists in the coupling algorithm and its properties. The Embedded Boundary method that we use is essentially identical to previous works [43,14,22,25]. The different versions of the Embedded Boundary method mainly differ in the way the stability condition arising from small cut-cells is enforced and we develop here a slightly different procedure in order to deal with solid boundaries coming close to each other. The method can be implemented independently from the time integration scheme used for the fluid, whether based on space–time splitting or multi-level time integration. Conservative fictitious domain methods have proven to give satisfactory conservation results for inviscid compressible flows in the case of static solid boundaries. Nevertheless, to our knowledge, conservation issues of the coupling have not been studied in the case of moving solids. We establish new conservation results in such a case. Our coupling method is designed to be capable of treating the general case of moving deformable bodies. In the present work, however, we only consider non-deformable (rigid) solid bodies. The case of deformable bodies is the object of ongoing work.

The fluid and solid solvers that we consider were chosen according to their ability to deal with shock waves and fracturing solids. The solid solver is based on a Discrete Element method, implemented in a code named Mka3D in the CEA [35]. It can handle elasticity as well as fracture and impact of solids. Solids are discretized into polyhedral particles, which interact through well-designed forces and torques. The particles have a rigid-body motion, and fracture is treated in a straightforward way by removing the physical cohesion between particles. The work reported in this article is a first step towards the coupling with the Mka3D code. The time integration scheme used by Mka3D (Verlet for displacement of the center of mass and RATTLE for rotation [38]) is retained for the rigid body treatment. Concerning the fluid solver, we use a Cartesian grid explicit finite volume method, based on the high-order one-step monotonicity-preserving scheme developed in [8] and space time splitting. However we emphasize that our coupling method is independent from both the Discrete Element method (as long as a solid interface is defined) and the numerical scheme used for the fluid calculation.

The article is organized as follows: we first present briefly the solid and fluid methods in Section 2. In sections 3 and 4, we describe the proposed explicit coupling procedure between the fluid and the moving solid in the framework of an Embedded Boundary method. The analysis of the conservation properties of the coupling is reported in Section 5, where we show that mass, momentum and energy of the solid–fluid system are exactly preserved. In Section 6, we demonstrate results about the preservation on a discrete level of two solid–fluid systems in uniform movement. Finally, we illustrate the efficiency and accuracy of the method on one and two-dimensional static and dynamic benchmarks in Section 7.