Journal of Computational Physics 283 (2015) 388–407

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A comparative study of penalization and phase field methods for the solution of the diffusion equation in complex geometries

Gerardo Tauriello, Petros Koumoutsakos *

Professorship of Computational Science, ETH Zürich, Clausiusstrasse 33, CH-8092 Zürich, Switzerland

ARTICLE INFO

Article history:
Received 11 July 2014
Received in revised form 19 October 2014
Accepted 25 November 2014
Available online 10 December 2014

Keywords:
Fictitious domain methods
Phase field methods
Volume penalization
Level set methods
Parabolic problems

ABSTRACT

We present a comparative study of penalization and phase field methods for the solution of the diffusion equation in complex geometries embedded using simple Cartesian meshes. The two methods have been widely employed to solve partial differential equations in complex and moving geometries for applications ranging from solid and fluid mechanics to biology and geophysics. Their popularity is largely due to their discretization on Cartesian meshes thus avoiding the need to create body-fitted grids. At the same time, there are questions regarding their accuracy and it appears that the use of each one is confined by disciplinary boundaries. Here, we compare penalization and phase field methods to handle problems with Neumann and Robin boundary conditions. We discuss extensions for Dirichlet boundary conditions and in turn compare with methods that have been explicitly designed to handle Dirichlet boundary conditions. The accuracy of all methods is analyzed using one and two dimensional benchmark problems such as the flow induced by an oscillating wall and by a cylinder performing rotary oscillations. This comparative study provides information to decide which methods to consider for a given application and their incorporation in broader computational frameworks. We demonstrate that phase field methods are more accurate than penalization methods on problems with Neumann boundary conditions and we present an error analysis explaining this result.

1. Introduction

The solution of partial differential equations (PDE) in complex and moving geometries is at the core of numerous application domains. The discretization of the governing equations can be broadly distinguished in Eulerian and Lagrangian approaches. We believe that a key distinction between these two approaches is the way that boundary conditions are enforced. For example, in solid mechanics, we compute the numerical solution of the equations of motion within a given physical domain while boundary conditions (BC) are defined either on the displacement of the interface or on the traction applied to it. Lagrangian or Arbitrary Lagrangian Eulerian methods track physical interfaces explicitly to simplify the handling of BC. While doing so, Lagrangian methods assume a fixed connectivity within the physical domain. This can lead to numerical errors in problems with extreme loadings and large deformations [1,2]. Eulerian methods, on the other hand, solve the equations on a fixed grid while the physical domain is deforming. Since such methods do not assume a fixed

* Corresponding author. Tel.: +41 44 632 52 58; fax: +41 44 632 17 03.
E-mail address: petros@ethz.ch (P. Koumoutsakos).

http://dx.doi.org/10.1016/j.jcp.2014.11.033
0021-9991/

connectivity, they can readily handle highly deforming solids. Furthermore, Eulerian methods with deforming and moving obstacles are commonly employed in fluid mechanics and they have been used for challenging simulations of single and multiple swimmers [3,4] and mixing devices [5]. Such methods can readily be coupled with Eulerian solid mechanics methods to enable the study of flow–structure interaction problems [6,7]. Similar problems with large deformations occur in numerical simulations of biological growth processes. There, one commonly requires the solution of PDEs to compute mechanical properties and the diffusion of chemical species within growing and deforming objects [8–11].

PDEs can efficiently be solved on regularly shaped domains by employing simple Cartesian grids. These grids facilitate the development of the numerical methods and their implementation on High Performance Computing architectures [12,13]. PDE solvers for problems with complex geometries usually employ carefully generated body-fitted meshes. The equations can then be discretized with a finite element method (FEM) [14] or a finite difference method suited for unstructured meshes [15]. Those methods are highly accurate but have two major drawbacks: First, the resulting computations are more irregular and expensive than for a regular mesh. Second, the mesh generation can be computationally expensive and this cost may become prohibitive for applications with moving and deforming domains even when Arbitrary Lagrangian Eulerian methods [16,17] are used.

Numerous works have addressed the solution of PDEs within complex geometries without a body-fitted mesh. We distinguish grid based and particle methods and present a non-exhaustive list of related works. For grid based methods, the list starts with the extensions of solutions of Poisson's equation from rectangular to irregular regions [18]. Hunt introduced a two dimensional (2D) finite difference method to approximate Robin BC without a body-fitted mesh by defining special discretizations of the spatial derivatives close to the boundary [19]. The classic immersed boundary method embeds the irregular boundaries in regular Cartesian meshes and adds a forcing term to the PDE close to the boundary to impose the BC [20,21]. The immersed interface method, proposed as a more accurate alternative to the immersed boundary method, translates the forcing term into jump conditions for the solution variable of the PDE [22]. Numerically, this again introduces special discretizations of the spatial derivatives close to the boundary which can be defined with finite difference, finite volume or finite element methods [23,24].

Level set methods were introduced as a powerful technique to accurately capture smooth interfaces and their deformations [25–28]. Based on the implicit level set representation of the interface one can define methods to extrapolate values from one side of the interface to the other one [29]. This can be used to solve PDEs within an irregular domain by embedding it into a regular domain and using "ghost" grid points near boundaries [30]. Such extrapolation methods can be computationally expensive as they require the solution of several hyperbolic PDEs at each time step to impose the BC. Alternatively, one can use the level set function to accurately identify the location of the interface and define special discretizations of the spatial derivatives close to the boundary for Dirichlet BC [31].

Phase field methods are an alternative to level set methods to implicitly define interfaces and have been used in various fields such as crystal growth, multi-component fluid flows and material sciences [32–34]. Commonly, the phase field is used to distinguish between phases in a material. Kockelkoren et al. introduced a variant of the phase field method to solve the diffusion equation within an irregular domain embedded in a larger regular domain [35]. Levine and Rappel later extended this method to impose Robin BC by adding a forcing term to the PDE [36]. This method was recently coupled with a remeshed particle level set method to solve reaction–diffusion systems on the surface and the interior of deforming geometries [11]. The "diffuse domain" approach extends the phase field method to prescribe Dirichlet, Neumann and Robin BC with a variety of forcing terms [37].

Penalization methods follow a similar approach by embedding the irregular domain into a larger regular domain called "fictitious domain" and adding a forcing term to the PDE. In contrast to immersed boundary methods, this forcing term may affect all of the "external" part of the larger domain. This is for instance used to impose Dirichlet BC for the Navier–Stokes equations by using Brinkman penalization [38–40]. Ramiere et al. proposed a "spread interface" penalization method to impose Dirichlet, Neumann and Robin BC for the solution of elliptic problems [41]. Kadoch et al. introduced a volume penalization method to impose homogeneous Neumann BC for scalar advection–diffusion with moving obstacles as in a chemical mixer [5]. Theoretical error estimates for fictitious domain and penalization methods were computed for solutions to Poisson's equation with flux BC [42].

Particle methods [43–45] have been proposed for the solution of PDEs such as diffusion equations, starting with the method of Particle Strength Exchange [46]. The incorporation of BC in such methods remains a topic of active research [11,47–49] with significant success in the formalism of the Reproducing Kernel Particle methods [50]. In recent years, it has become evident that particle methods simulating mechanical systems with large deformations need to be coupled with remeshing procedures [51,52] that employ a Cartesian grid. Hence, the enforcement of BC in remeshed particle methods can be mapped back to the problem of enforcing them on a Cartesian grid.

In this work, we consider the diffusion equation as the model PDE and use it to compare solution methods for irregular domains that employ Cartesian grids for low computational cost [5,31,35,36,39,41]. To the best of our knowledge, those methods were never carefully compared on identical time dependent test problems and we believe that this comparison can guide decisions on which methods to consider for a given problem.

We exclude immersed boundary and immersed interface methods from our analysis as they cannot readily be used to solve the diffusion equation within a given domain. They have been designed to solve flow problems with an interface within the flow as opposed to solving a general PDE within a domain. We furthermore exclude methods which require ghost values near boundaries as they require computationally expensive extrapolations to compute those ghost values. The