Journal of Computational Physics 354 (2018) 692–716

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A Vortex Particle-Mesh method for subsonic compressible flows

Philippe Parmentier *, Grégoire Winckelmans, Philippe Chatelain

Institute of Mechanics, Materials and Civil Engineering, Université catholique de Louvain, 1348 Louvain-la-Neuve, Belgium

ARTICLE INFO

Article history:
Received 23 January 2017
Received in revised form 22 September 2017
Accepted 23 October 2017
Available online 21 November 2017

Keywords:
Vortex Particle-Mesh methods
Compressible flow
Subsonic aeroacoustics
Unbounded domain
High-order methods

ABSTRACT

This paper presents the implementation and validation of a remeshed Vortex Particle-Mesh (VPM) method capable of simulating complex compressible and viscous flows. It is supplemented with a radiation boundary condition in order for the method to accommodate the radiating quantities of the flow. The efficiency of the methodology relies on the use of an underlying grid; it allows the use of a FFT-based Poisson solver to calculate the velocity field, and the use of high-order isotropic finite differences to evaluate the non-advective terms in the Lagrangian form of the conservation equations. The Möhring analogy is then also used to further obtain the far-field sound produced by two co-rotating Gaussian vortices. It is demonstrated that the method is in excellent quantitative agreement with reference results that were obtained using a high-order Eulerian method and using a high-order remeshed Vortex Particle (VP) method.

1. Introduction

Vortex particle (VP) methods are known to be well suited for the study of convection dominated incompressible flows in an unbounded domain. This capability is attributable to different interesting properties either of the mathematical formulation itself or of the efficient implementations that follow from it. First, VP methods are Lagrangian methods solving the Navier–Stokes equations in vorticity–velocity formulation [1,2]. Particles are used to discretize the vorticity field while being advected by the local velocity field. This Lagrangian treatment of the advection, combined with high-order remeshing schemes – that are required to compensate for the Lagrangian distortion of the particles – has two expected advantages: relaxed stability constraints and a reduced dispersion error. Besides, because of their formulation, VP methods are inherently capable of taking into account unbounded boundary conditions. They also greatly benefit from the compactness of the vorticity field. Vortex Particle-Mesh (VPM) methods, also referred to as Vortex-in-Cell (VIC) methods, derive from it. The Vortex-in-Cell method was first introduced by Christiansen [3] and additional developments were made by others (e.g. [1,2] as well as references therein and [4–6]). While keeping the advantages of VP methods, VPM methods also exploit the advantages of an underlying grid: it allows the use of efficient grid-based Poisson solvers (that are needed to calculate the velocity field) and the use of high-order finite difference operators to evaluate the non-advective terms. The high quality of the remeshed VPM methodology was demonstrated against the results obtained using a pseudo-spectral solver in [7]. The advantages of VP and VPM methods were exploited in numerous studies, both in 2-D and in 3-D, and efficient im-

* Corresponding author.
E-mail addresses: p.parmentier@uclouvain.be (P. Parmentier), gregoire.winckelmans@uclouvain.be (G. Winckelmans), philippe.chatelain@uclouvain.be (P. Chatelain).

https://doi.org/10.1016/j.jcp.2017.10.040
0021-9991/

plementations were already achieved on massively parallel architectures both in the case of CPUs and GPUs [8–11]. The major challenges of the VPM method and vortex methods in general most probably reside in the enforcement of the zero-divergence of the vorticity field in 3-D simulations, on the one hand, and in the treatment of wall-bounded flows, on the other hand. For further and extensive information on vortex methods, one can refer to the book of Cottet and Koutmoutsakos [1] and the chapter of Winckelmans in [2], as well as to the references therein.

When compressibility effects can be neglected, numerous studies demonstrated the validity of the strategy of using an incompressible flow solver in conjunction with acoustic analogies so as to predict the far field sound. Of course, vortex methods have been no exception (e.g. Knio et al. [12], Pothou et al. [13], Jeon et al. [14] and the extensive review of Colonius and Lele [15]) due to their ability to accurately follow the evolution of the vorticity field, which is of primary importance in sound generation. We also note the work of Huberson et al. in the context of helicopter noise [16]. The strategy implemented consisted in employing a VPM methodology to solve the three-dimensional unsteady inviscid and incompressible flow past a body, the results of which being used as an input for the acoustic wave equation. A section-by-section 2-D compressible Euler correction was used when the compressibility effects could not be neglected (e.g. on the advancing blade).

Some efforts have already been made to bring the Vortex Particle method to the compressible regime: see e.g. the review of Winckelmans [2], the paper of Eldredge et al. [17], and references therein. We will here focus on the VP method for fully compressible flows of Eldredge et al. [17] and the existing VPM methods for compressible flows [18–20] as they are the most closely related to the methodology proposed in this paper.

Eldredge et al. [17] developed a high-order remeshed VP method devoted to the simulation of two-dimensional compressible and viscous flows. It will be here referred to as the DVP solver, which stands for Dilating Vortex Particle solver, as in the original paper. The method follows the evolution of integrated quantities over a material volume of the vorticity, $\boldsymbol{\omega} = \nabla \times \mathbf{u}$, (as in an incompressible flow solver), the dilatation (i.e. the divergence of the velocity field, $\theta = \nabla \cdot \mathbf{u}$), the density, the enthalpy and the entropy fields. The Helmholtz decomposition is used to compute the velocity field, which is induced by the vorticity field, and the dilatation field, and with the further addition of an imposed far-field potential flow in some cases. Solving for velocity in this Helmholtz decomposition entails the resolution of Poisson equations. In [17], those are solved using a Fast Multipole Method (FMM) [21] adapted to also take into account the dilatation. The DVP solver also relied on an extension of Particle Strength Exchange (PSE) schemes to allow the high-order evaluation of arbitrary spatial differential operators [22]. One-sided stencils were also developed and allowed the enforcement of the required nonreflecting boundary condition needed to truncate the unbounded domain. Eldredge et al. used an eighth-order accurate kernel in the interior of the domain and a second-order accurate kernel to compute the one-sided derivatives at the boundaries. A sixth-order template function was used as the blob function and also as the particle redistribution kernel. This high-order kernel also exhibited a filtering capability. Those two features were necessary to preserve the smoothness of the enthalpy field during the remeshing procedure. The DVP solver was used to simulate several complex fully compressible flows; the baroclinic generation of vorticity, the sound generation by a co-rotating vortex pair [17]. The resulting scheme was shown to be consistent and in very good agreement with the results of Mitchell et al. [23], who used an Eulerian solver with sixth-order Padé scheme to compute the spatial derivatives. The DVP solver was also used to carry out a thorough analysis of the dynamics and acoustics of viscous 2-D leapfrogging vortices [24].

Regarding the VPM methodology, Thirifay and Winckelmans developed a remeshed VP method dedicated to the study of reacting flows with heat release (and combustion problems especially), using the weakly compressible approximation [25, 2]; they also introduced mesh-based solvers to make it a VPM method referred to as a Vortex-in-Cell (VIC) method by the authors [18].[1] In the weakly compressible case and contrarily to the fully compressible case, the compactness of the fields – one of the most attractive features of vortex methods – can still be usefully exploited.

Following the work of Oxley [19], Papadakis and Voutsinas [20] coupled a low-order VPM solver for compressible flows with a near-body Eulerian code solving the compressible Unsteady Reynolds-Averaged Navier–Stokes (URANS) equations. As it intrinsically and conveniently allows the use of a non-isotropic grid, the Eulerian solver is responsible for the near-body region, while the Eulerian–Lagrangian VPM method is exploited in the wake. The Poisson problems with unbounded boundary conditions were dealt with using a solver [26] based on the James–Lackner algorithm [27,28] and the Method of Local Corrections [29,30]. The viscous terms were accounted for using a PSE scheme on the grid. The solver was parallelized using a noniterative domain decomposition technique [26]. The particles were periodically remeshed using the third-order accurate M_4' scheme.

The PSE-like methodology exploited in [17] replaces the differential operators by integral operators that eventually require a quadrature to be applied over the particles. Such an implementation is based on nearest neighbors finding and is thus less computationally efficient than a mesh-based evaluation. The latter is adopted in the present work as it allows the use of an underlying grid giving access to efficient grid-based solvers and operators.

Additionally, Eldredge et al. used for their redistribution a high-order and large support interpolation kernel that included some degree of smoothing. In this paper, we wish to pursue an approach in which we regain separate controls over the explicit filtering of fields and the particle remeshing procedure, which does involve some implicit smoothing; this allows more flexibility for operations subject to different constraints on their application frequency. It also allows to benefit from recent advances in high-order interpolation kernels and in high-order grid-based filters.

[1] For the sake of clarity, we here stick with the VPM naming.