Post-processing of two-phase DNS simulations exploiting geometrical features and topological invariants to extract flow statistics: application to canonical objects and the collision of two droplets

Ruben Di Battista, Iván Bermejo-Moreno, Thibaut Ménard, Stéphane de Chaisemartin, Marc Massot

Post-processing of two-phase DNS simulations exploiting geometrical features and topological invariants to extract flow statistics: application to canonical objects and the collision of two droplets

Ruben Di Battista, Iván Bermejo-Moreno, Thibaut Ménard,
Stéphane de Chaisemartin and Marc Massot

École Polytechnique, Centre de Mathématiques Appliquées

Palaiseau, 91128 CEDEX, France

University of Southern California

Los Angeles, CA 90007, United States

CORIA

Avenue de l'Université, 76801 Saint-Étienne-du-Rouvray, France

IFPEN

Avenue du Bois Préau, 92852, Rueil-Malmaison, France

ruben.di-battista@polytechnique.edu

Keywords: DNS, geometrical, post-processing, curvature

Abstract

This work presents a methodology to collect useful flow statistics over DNS simulations exploiting geometrical properties maps and topological invariants. The procedure is based on estimating curvatures on triangulated surfaces as as averaged values around a given point and its first neighbours (the 1-ring of such a point). In the case of two-phase flow high-fidelity simulations, the surfaces are obtained after an iso-contouring procedure of the volumetric level-set field. The estimation of the curvatures on the surface allows the possibility of characterizing the 3D objects that are created in a high-fidelity simulation in terms of their area-weighted geometrical maps. In this work we provide an assessment of the robustness of the curvature estimation algorithm applied to some canonical 3D objects and to the Direct Numerical Simulation of the collision of two droplets. We provide the tracking of the topological evolution of such objects in terms of geometrical maps and we highlight the effect of mesh resolution on those topological changes.

1 Introduction

Classically two-phase flow modeling has been addressed with reduced-order models because generally, even in modern times, the cost of performing high-fidelity simulations is prohibitive for configuration even mildly more complex than academic benchmark cases. A classical work on the subject is Ishii et al. (1984) where the multiphase flow equations are derived through an averaging process performed on the instantaneous phase flow equations. Another important result in the frame of two-phase flow modeling is presented in Baer et al. (1986), where the entropy inequality is enforced on the equations system providing a problem that is mathematically well-posed in the context of simulation of solid propellants combustion. Gavrilyuk et al. (1999) rediscover the variational approach to derive models for two-phase flow simulation. In Baer et al. (1986), the procedure to derive equations does not explicitly highlight the convective part of the problem *w.r.t.* the dissipative part as it's done in Gavrilyuk et al. (1999) and Drui et al. (2016). In particular, in the work of Drui et al., a hierarchy of two-phase flow models is retrieved applying a variational approach that clearly provides as outcome the convective part of the system. The dissipative part is then retrieved again enforcing the entropy inequality, clearly depicting the roles of the different parts composing the system. In Drui et al. (2016), an energy term responsible for the pulsation of bubbles in the unresolved scale (subscale energy term) is proposed. In these modeling efforts, geometrical configuration of the interface between the two phases is generally described only through the volume fraction, while richer approaches could be beneficial to simulation fidelity. A first work on the matter has been proposed by Pope (1988)

and Drew (1990), where governing equations for geometrical parameters are given in the context of flame front propagation. Moreover Essadki et al. (2017) have shown that intrinsic topological relation as the Gauss-Bonnet theorem can be used to rely different regimes in a two-phase flow system (with particular interest to injector systems) allowing the connection between the Surface Density Function (SDF), *i.e.* a statistical description locally to the resolved interface, to the Number Density Function (NDF) in the disperse-phase region of an injection system. A first example of a reduced model for two-phase flows that contains additional geometrical contributions is reported in Cordesse et al. (2018), where the Least Action Principle (LAP) is used to obtain a set of equations that include a macroscopic term and a microscopic (subscale) term linked to surface tension. Most of the times introducing additional richness to the geometrical description of the interface brings unclosed terms that need closure, the work described in this paper is motivated by the need of developing such closure. The characterization of 3D objects in terms of geometrical features in high-fidelity simulations is not new in literature, even though, at the best of our knowledge, we did not found other studies directly applied to injections problems. Mean and Gauss curvatures are not the only way to parameterize the phase space of interest, in facts Lewiner et al. (2003) introduces two non-linear functions of the curvatures above mentioned, *Curvedness* and *Shape Index*, that are then used in Bermejo-Moreno and Pullin (2008) and Bermejo-Moreno, Pullin, and Horiuti (2009) to provide a throughout analysis and characterization of objects within turbulent flow. In Leung et al. (2012) *Minkowski functionals* are used instead to perform a similar analysis. Dopazo et al. (2018) track iso-surfaces in homogeneous turbulence also in terms of curvature value. The prominent flow contributions to the flame curvature generation are highlighted in Cifuentes et al. (2018) within the context of the combustion in a bluff body burner.

In this study we propose the statistical characterization of two canonical objects of which analytical expression for the curvatures are available (a sphere and an ellipsoid) in order to assess the robustness and reliability of the computational strategy. We then study the Direct Numerical Simulation (DNS) simulation of two droplets colliding against each other in terms of geometrical properties. In section 2 we present the numerical algorithms needed to compute curvature values no the discrete surface triangulation: we present a way of computing these values as 1-ring average values as proposed by Meyer et al. (2003). In section 3 then we analyze the evolution of the geometrical statistical maps while the simulation progresses in order to catch interesting correlations or patterns that could be useful to formulate closures for unknown terms or to enrich existent CFD models and we furthermore clarify how the algorithms behave in term of intrinsic constraints preservation. Section 4 will be reserved to our final remarks.

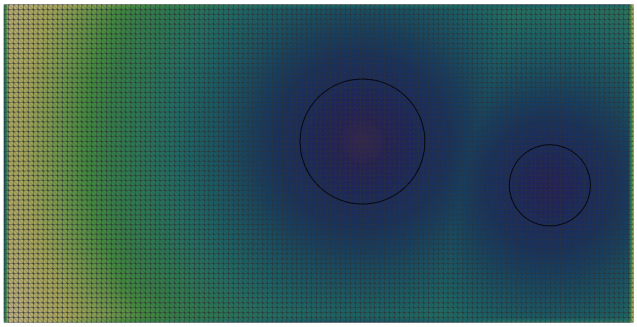


Figure 1: A central slice at $\tau = 0$ of the volumetric level-set field on a uniform grid. The black lines are the 0-isoline of the level-set function

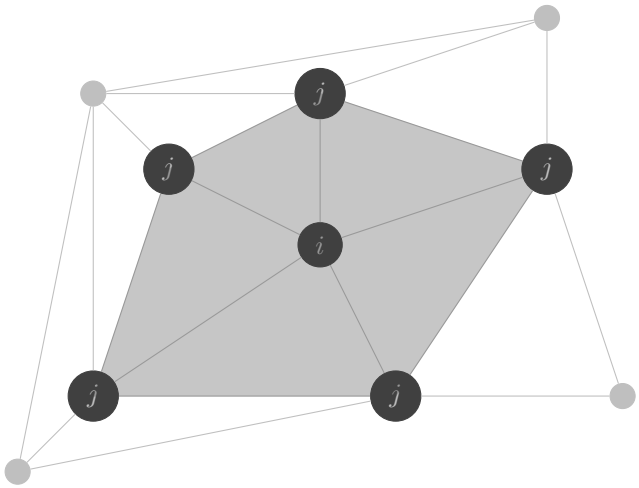


Figure 2: The 1-ring of a point $\boldsymbol{x}_i$

2 Geometrical properties as 1-ring averaged values

In this section we describe the numerical strategies used to estimate the geometrical properties on a triangulated surface. The triangulated surface is obtained applying a contouring procedure (Schroeder, Maynard, et al. (2015)) to the initial level-set that is available on the entire volume. The DNS simulations on which we apply these estimations are performed using the `ARCHER` code from the CORIA lab Ménard et al. (2007). The volumetric level-set field is given on a regular uniform rectangular grid as the one shown in the slice of fig. 1. We must note that the quality of the estimations are influenced not only by the quality of the strategy chosen to approximate the topological parameters, but also on the quality of the triangulation. Even though the Marching Cubes (MC) algorithm (with its variants as the Flying Edges (FE) algorithm used here) is a well-established method to obtain isolines of a field in a volume, there exist also more advanced revisions (Lewiner et al. (2003)) of the MC algorithm that ensure additional topological guarantees that could improve the initial estimates for the needed quantities.

The algorithm presented in this section is based on Meyer et al. (2003). The idea is to approximate the value of the mean curvature H and the Gauss curvature G on a vertex

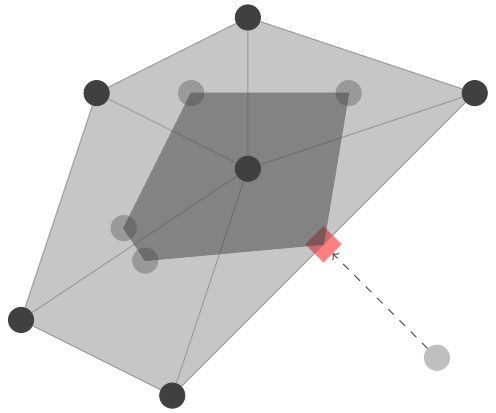


Figure 3: A_{mixed} for a general case where one of the triangles is obtuse. The light gray area is the 1-ring neighbourhood and the dark gray area is actually A_{mixed}

of the triangulated surface with the average value computed around the 1-ring neighbourhood of said vertex. The 1-ring neighbourhood of a point $\boldsymbol{x}_i$ is the region included in the set of points $\boldsymbol{x}_j$ that are directly connected to $\boldsymbol{x}_i$, fig. 2 shows a graphical representation. The other quantities are derived from H and G. This strategy of computing geometrical properties has been implemented in a GPL-licensed library called `Mercur(v)e` (Di Battista (2018)).

In order to compute the mean curvature value, the algorithm leverages a local to the 1-ring neighbourhood discretization of the Laplace-Beltrami (LB) operator whose continuous expression is reported in eq. (1).

$$\mathcal{LB}(\boldsymbol{x}) = 2H(\boldsymbol{x})\hat{\boldsymbol{n}}(\boldsymbol{x}) = \boldsymbol{K}(\boldsymbol{x}) \tag{1}$$

The LB operator can be approximated on a triangulated surface via eq. (2).

$$\overset{\circ}{[\![\boldsymbol{K}]\!]}(\boldsymbol{x}_i) = \frac{1}{2A_{\text{mixed}}} \sum_{j}^{\text{1-ring}} (\cot\alpha_{ij} + \cot\beta_{ij})(\boldsymbol{x}_i - \boldsymbol{x}_j) \tag{2}$$

With $\overset{\circ}{[\![\bullet]\!]}$ we denote the discrete estimation of a certain quantity on the mesh averaged around the 1-ring. Once the discrete LB has been computed, the scalar value of the mean curvature is simply the half of the norm of the LB vector:

$$\overset{\circ}{[\![H]\!]} = \frac{1}{2}\left|\overset{\circ}{[\![\boldsymbol{K}]\!]}\right| \tag{3}$$

A_{mixed} is the equivalent 1-ring area: it is equal to the Voronoi region area if all the triangles that compose the 1-ring are non-obtuse, otherwise is the region whose perimeter is given by all the circumcenters of the non-obtuse triangles and the midpoints of the edges opposed to the obtuse angles for the triangles that are obtuse. See fig. 3 for clarity. The actual algorithm to compute A_{mixed} is given in Meyer et al. (2003), and it is reported in algorithm 1 for reference. What is interesting is to note that the area for a Voronoi region (*i.e.* the area of the polygon given by all the circumcenters of the triangles composing the 1-ring, all of them being non-obtuse) is given by eq. (4), where the $(\cot\alpha_{ij} + \cot\beta_{ij})$ term can be reused successively in eq. (2), saving up on computational cost.

$$A_{\text{voronoi}} = \frac{1}{8}\sum_{j}^{\text{1-ring}} (\cot\alpha_{ij} + \cot\beta_{ij})\,|\boldsymbol{x}_i - \boldsymbol{x}_j|^2 \tag{4}$$

Algorithm 1 Algorithm to compute the mixed area

procedure MIXEDAREA($\boldsymbol{x}_i$) ▷ We compute the A_{mixed} on the 1-ring of $\boldsymbol{x}_i$
  $A_{\text{mixed}} = 0$
  for T in ONERING($\boldsymbol{x}_i$) **do** ▷ For each triangle composing the 1-ring
    $\widehat{\boldsymbol{x}}_i, \widehat{\boldsymbol{x}}_{ij}, \widehat{\boldsymbol{x}}_{ik} \leftarrow$ ANGLES(T) ▷ $\boldsymbol{x}_{ij}$ and $\boldsymbol{x}_{ik}$ are the 1-ring points connected to $\boldsymbol{x}_i$
    $A \leftarrow$AREA(T)
    if $\cot\widehat{\boldsymbol{x}}_{ij} > 0$ or $\cot\widehat{\boldsymbol{x}}_{ik} > 0$ **then**
      A_{mixed} += $A/4$
    else if $\cot\widehat{\boldsymbol{x}}_i > 0$ **then**
      A_{mixed} += $A/2$
    else
      A_{mixed} += $\frac{1}{8}|\boldsymbol{x}_i - \boldsymbol{x}_{ik}|^2 \cot\widehat{\boldsymbol{x}}_{ij} + |\boldsymbol{x}_i - \boldsymbol{x}_{ij}|^2 \cot\widehat{\boldsymbol{x}}_{ik}$

In order to compute the value of the Gauss curvature G we can just rely on the definition of G for a geodesic polygon:

Definition 1 *given a geodesic polygon close to a point $\boldsymbol{x}_i$ of area A, the product of the Gauss curvature times the area of the polygon is equal to 2π minus the defect of the exterior angles ϵ_k of the polygon*

$$G = \frac{1}{A}\left(2\pi - \sum_k \epsilon_k\right) \tag{5}$$

Since for the triangulation of a surface the 1-ring neighbourhood of a point $\boldsymbol{x}_i$ is actually a special case of a geodesic polygon for which the edges are straight lines instead of geodesics, the same relation can be recasted for the internal angles θ_j, as shown in eq. (6).

$$\overset{\circ}{[\![G]\!]} = \frac{1}{A_{\text{mixed}}}\left(2\pi - \sum_j^{\text{1-ring}} \theta_j\right) \tag{6}$$

Once the mean and Gauss curvature are calculated, the other typical differential geometry parameters can be retrieved from the values of H and G. For example the principal curvatures k_1 and k_2 are computed as follows:

$$k_1(\boldsymbol{x}_i) = \overset{\circ}{[\![H]\!]}(\boldsymbol{x}_i) + \sqrt{\overset{\circ}{[\![\Delta]\!]}(\boldsymbol{x})} \tag{7}$$

$$k_2(\boldsymbol{x}_i) = \overset{\circ}{[\![H]\!]}(\boldsymbol{x}_i) - \sqrt{\overset{\circ}{[\![\Delta]\!]}(\boldsymbol{x})} \tag{8}$$

As we will see later in section 3, this strategy allows to provide estimates on the triangulation of 3D objects that are quite smooth *w.r.t.* to other strategies. A problem that arises, unfortunately, is linked to the fact that H and G are calculated independently one from the other. That means in particular that $\Delta = H^2 - G$, that in the continuous case is always ≥ 0, in the discrete case can be < 0. This problem is mentioned in Meyer et al. (2003) to be an "extremely rare occurrence", while in our computations this happens quite often.

3 Results

In this section we present some results obtained exploiting the numerical strategies to compute geometrical parameters that are described in section 2. First of all in section 3.1 we benchmark the accuracy and convergence rate of the methods on some canonical objects for which the expressions of the curvatures in Cartesian coordinates are available. These 3D objects are generated using superquadrics primitive from VTK library (Schroeder, Martin, et al. (2006)). We then apply the computation to a DNS and we compute an area-weighted Probability Density Function (PDF) to highlight interesting footprints in the $H - G$ phase space of the topological objects produced at each simulation time step.

3.1 Validation on canonical 3D objects

In order to assess the validity and reliability of the algorithm we tested the curvatures computation on a sphere of radius R and on an ellipsoid of semi-axes a, b, c respectively for which analytical expression of curvatures in Cartesian coordinates are available analytically (eqs. (9) and (10), Bektas (2017)).

$$\begin{aligned} G_{\text{sphere}}(\boldsymbol{x}) &= 1/R^2 \\ H_{\text{sphere}}(\boldsymbol{x}) &= 1/R \end{aligned} \tag{9}$$

$$\begin{aligned} G_{\text{ell}}(\boldsymbol{x}) &= \frac{1}{\left[abc\left(\frac{x^2}{a^4} + \frac{y^2}{b^4} + \frac{z^2}{c^4}\right)\right]^2} \\ H_{\text{ell}}(\boldsymbol{x}) &= \frac{x^2 + y^2 + z^2 - a^2 - b^2 - c^2}{\left[2\,(abc)^2\left(\frac{x^2}{a^4} + \frac{y^2}{b^4} + \frac{z^2}{c^4}\right)\right]^{\frac{3}{2}}} \end{aligned} \tag{10}$$

The 3D objects are generated internally without any contouring procedure in order to provide high-quality triangulation *w.r.t.* MC-contoured level-sets (as generally done for full DNS), with the idea of separating effects due to the calculation of curvatures from the ones associated to triangulation quality. Two examples of triangulated objects are shown in fig. 4.

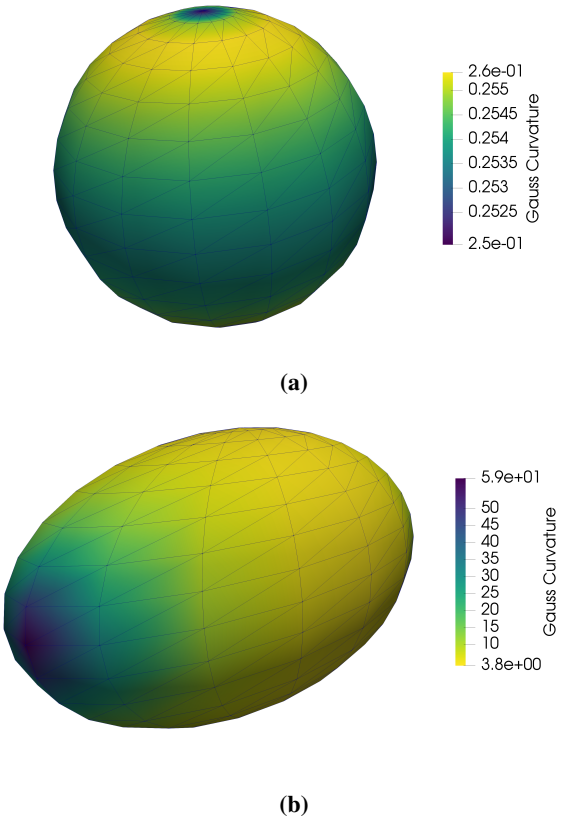


Figure 4: Examples of triangulated objects used in this section

In fig. 5a and fig. 5b we display the convergence rate for the two objects at increasing number of points. It is possible to notice that for both objects the algorithm computes the values of the two curvatures with reasonable precision and with a stable convergence rate. It is interesting to note how the convergence plot has a minimum around 10^5 points for both cases. We will show that the limit is related to float arithmetic bounds associated to very small cells that progressively start to appear on the object near the poles — this behaviour is associated to the way VTK library uses to build superquadrics surfaces —, caused by the trigonometric functions (notably the $\cot(\alpha)$) that are used to approximate 1-ring averaged H and G in Meyer et al. (2003). In addition, the error level for H and G for the ellipsoid case (fig. 5b) are different. This is due to the fact that the two parameters are estimated independently, without any guarantee on the coupled committed error. For this same reason, as we introduced in section 2, the algorithm produces some points for which $\Delta \not\geq 0$; fig. 7 shows the percentage of outliers respectively for the sphere (7a) and the ellipsoid (7b). In the first case, outliers are a big part of the total points, spanning from $\sim 23\%$ to $\sim 16\%$ of the total, while for the ellipsoid outliers are way less ($< 2\%$ for most of the resolutions). This behaviour is linked to the fact that for a sphere, on each point of the surface, $\Delta = 0$ exactly, hence all the analysis is played along the boundary of the admissible curvature values, and the algorithm is not able to cope with this constraint. Nonetheless in fig. 6 an area-weighted error distribution of the relative er-

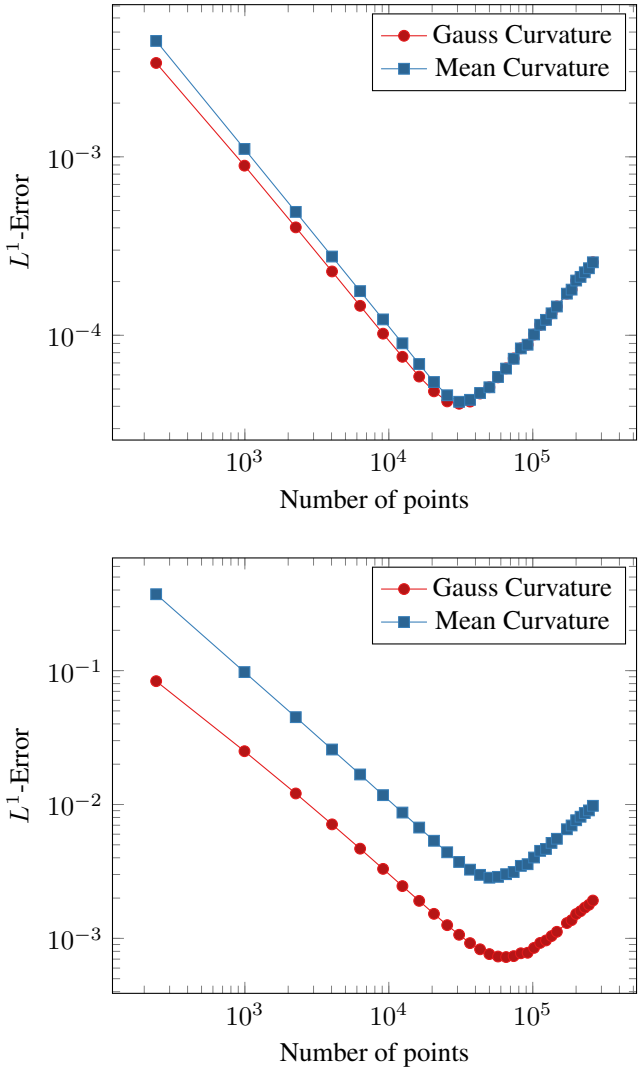


Figure 5: Convergence rate for H and G: (5a) for the sphere. (5b) for the ellipsoid

ror $E = (H_i - \sqrt{G})/\sqrt{G}$ is represented. The zones with negative relative error are associated to the outliers.

In order to have a better understanding on the error behaviour and the influence of the cell size, we computed two bi-variate PDFs, with and without area-weighting, that are reported in fig. 8 and fig. 9. The area-weighting means that each sample of the distribution is accounted for in the PDF taking into account its associated cumulative 1-ring area in such a way that small triangles are less important in *w.r.t.* big triangles. Figure 6 is indeed an integrated version of the PDF shown in figs. 8 and 9.

$$\mathcal{P}(E) = \int_A \mathcal{P}(A, E)\, \mathrm{d}A \tag{11}$$

The zone for which the error E is negative are the zones of points that do not respect the $\Delta \geq 0$ constraint. The bigger contributions to the error are given by points that have a relatively big 1-ring area associated to them and, hence, a less precise approximation of the curvature values. In addition, these "big" 1-rings are associated to outliers, since the associated error is always negative. It is worth to note, in addition, that increasing the resolution of the object and consequently reducing the area of the smallest triangles, triggers the generation of error associated to the loss of precision of floating point arithmetic: as we can see, in fig. 8, that is for a coarsely resolved sphere, the error distribution in the space (A, E) is substantially a one-dimensional sub-manifold, for all the surface areas, while for fig. 9, in the zones with very small triangles, the error is populated in all directions, that is with a different propagation mechanism than the one associated to discretization accuracy. This error associated to floating point precision loss explains the rising, after a certain resolution scale, of the L1-error for both curvatures that is shown in fig. 5a: small cells become too small to compute the curvature estimation and they add up to the global error.

3.2 Computation on DNS

Once we assessed the capabilities of our strategy on well-triangulated objects, we chose to perform the curvatures estimation on a representative two-phase flow benchmark case: the collision of two droplets. Two droplets with a radius respectively of $130\,\mu\mathrm{m}$ and $200\,\mu\mathrm{m}$ are configured at the beginning of the simulation with a negative relative velocity one *w.r.t.* to the other. The two spherical objects, then, start to approach until they collide causing surface deformation and topology changes. The simulation of the droplets collision is a DNS performed using the `ARCHER` code Ménard et al. (2007). In order to investigate the influence of the mesh resolution on the topological configurations we used three different mesh refinement levels: a coarse configuration with $128 \times 128 \times 256$ cells, a medium case with $256 \times 256 \times 512$ cells and a fine case with $512 \times 512 \times 1024$ cells. We characterize the footprint of each topological object with an area-based PDF in the phase space $H - G$, $\mathcal{P}(H, G)$: $A(H, G)$ is the area associated to a point in the phase space and it is used as a weighting function for the PDF. This PDF is computed on the data after filtering out outliers for which $\Delta < 0$

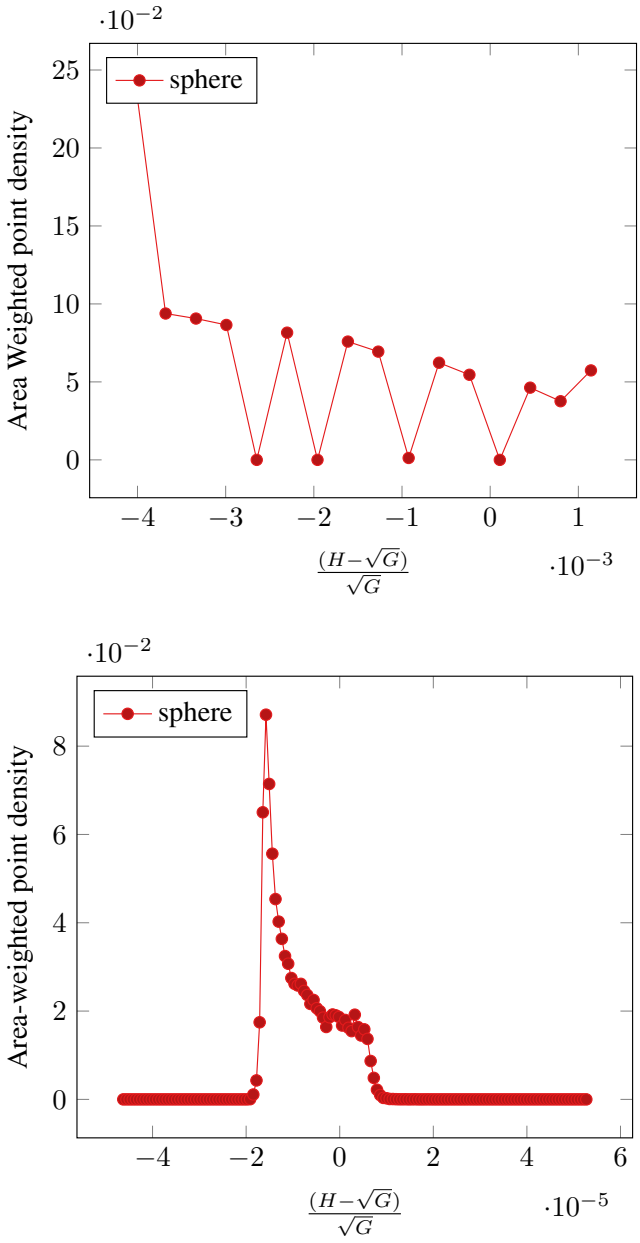


Figure 6: Error distribution for the sphere outliers. (6a) coarsest resolution. (6b) finest resolution

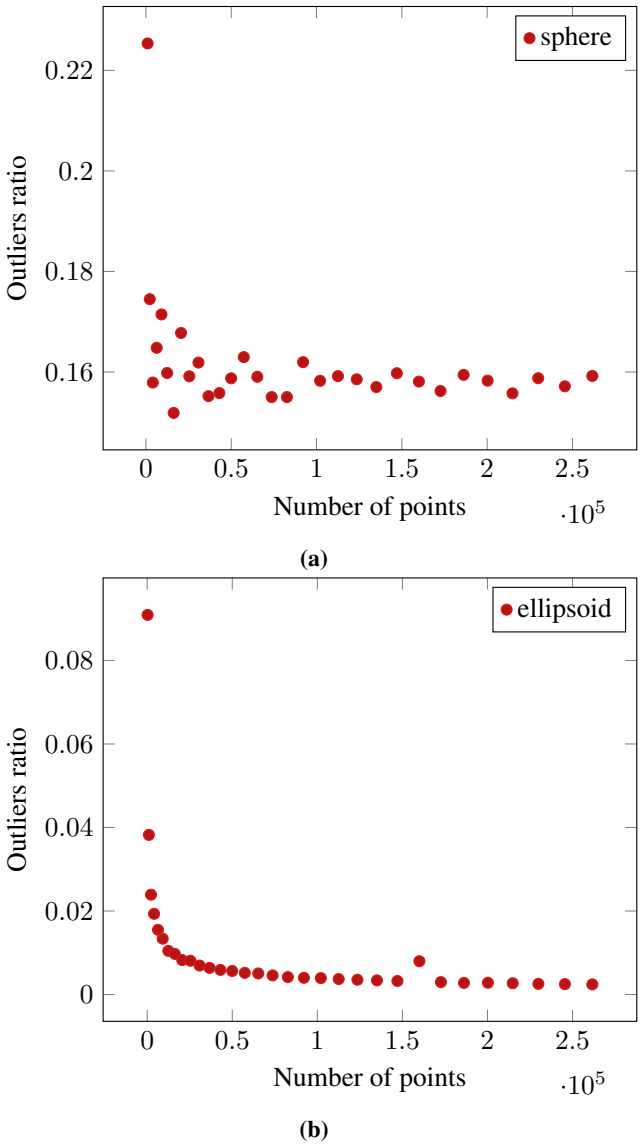
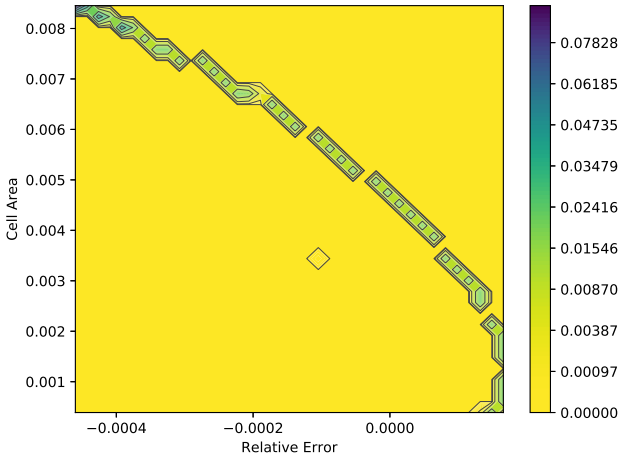
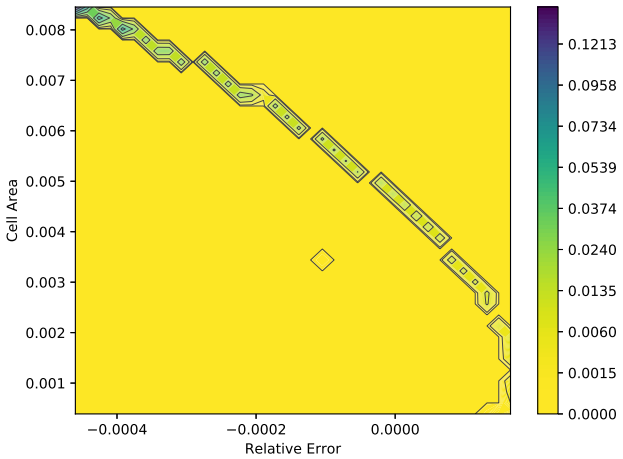


Figure 7: Percentage of outliers per total number of points. (7a) for the sphere and (7b) for the ellipsoid

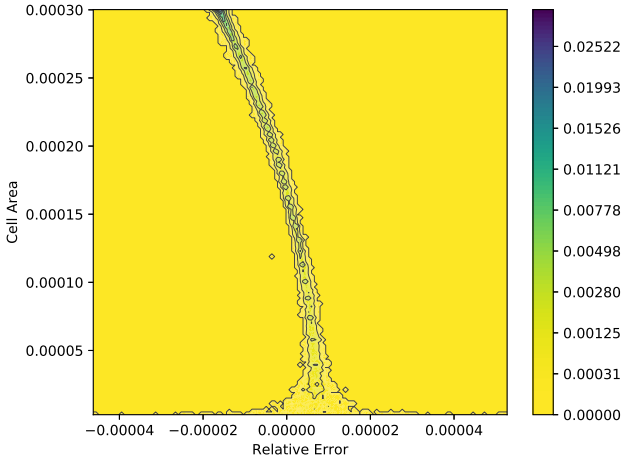


(a)

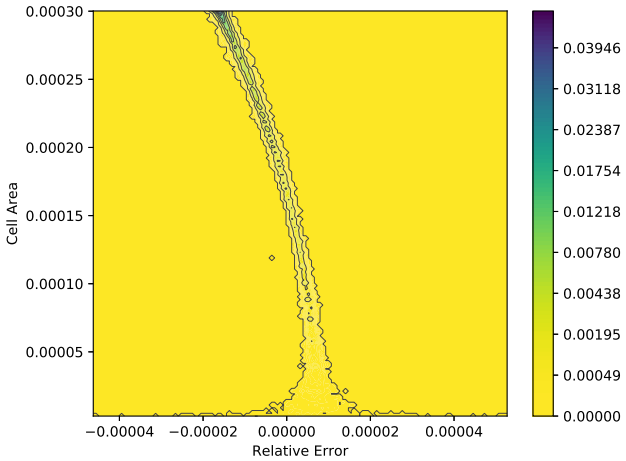


(b)

Figure 8: Bi-variate probability density functions for a coarse sphere. Figure 8a is a simple number density. While fig. 8b is an area-weighted PDF



(a)



(b)

Figure 9: Bi-variate probability density functions for a fine sphere. Figure 9a is a simple number density. While fig. 9b is an area-weighted PDF

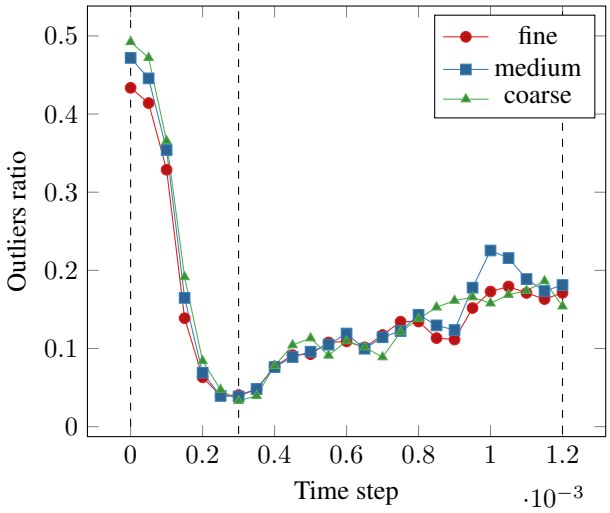


Figure 10: Evolution over time of the percentage of outliers. The black dashed lines are the snapshot displayed in figs. 11 to 13

with a number of bins $n \sim \sqrt{N}$ being N the total number of points in the surface. The corresponding values of H, G are normalized with the equivalent curvatures defined as in eq. (12).

$$\begin{aligned} R_{\text{eq}} &= \frac{R_1 + R_2}{2} \\ H_{\text{eq}} &= \frac{1}{R_{\text{eq}}} \\ G_{\text{eq}} &= \frac{1}{R_{\text{eq}}^2} \end{aligned} \tag{12}$$

We performed the outliers statistics analysis also in the droplets collision case. In fig. 10 we can appreciate again how outliers are more probably produced when the topological objects resemble a sphere, in facts at the beginning of the simulation, when we have two exact spheres in the domain, the outliers are large in absolute terms, even if for the finest case they are percentually less than the other cases. Once the objects is deformed after the collision, the outliers percentage steeply drops and stays averagely the same along the simulation. It's interesting to highlight how the number of outliers is not influenced by the triangulation refinement level but it's quite exclusively associated to the intrinsic topological configuration of the object: more the object is closer to a sphere, higher the number of outliers. We underline that, at least in this specific case, the outliers distribution is not really improved going from a coarse triangulation to a fine one.

A general remark we can provide is that the topological configurations at the same time-step are strongly influenced by the mesh refinement level: in multiphase flow, differently from what it happens in single-phase flow where a minimal scale can be defined — *i.e.* the Kolmogorov scale —, we do not have a minimal scale that, once reached with the numerical simulation, allows us to resolve everything with a DNS. That means we are always going to commit some sort of error that lies in the fact we are cutting artificially the resolution (due to computational resources) and for this reason we will be always under-resolving something at a certain scale. In addition to that the physics is strongly influenced by this (for example in the simulation at hands, producing a different number of satellite droplets depending on the case resolution). We can infer this for example comparing fig. 13b with figs. 14a and 14b: more the case is refined, smoother is the PDF that is extracted from the dataset. This behaviour can be related in this specific case to the precision which the surface tension, that depends on the mean curvature, is computed with. The more the mesh is resolved, the more accurate is the surface tension force, directly influencing the topological configuration.

In figs. 11 to 13 the PDF $\mathcal{P}(H, G)$ is shown for the different time steps compared to the topological configuration of the system. At the beginning of the simulation (fig. 11b), two points are visible on the phase space associated to the constant values of H/H_{ref} and G/G_{ref} for the two spherical particles (H/H_{ref}, 1.26, 0.825) , (G/G_{ref}, 1.61, 0.68). At the moment of the collision fig. 12b an object with lower values of curvatures is produced. In the following time steps the

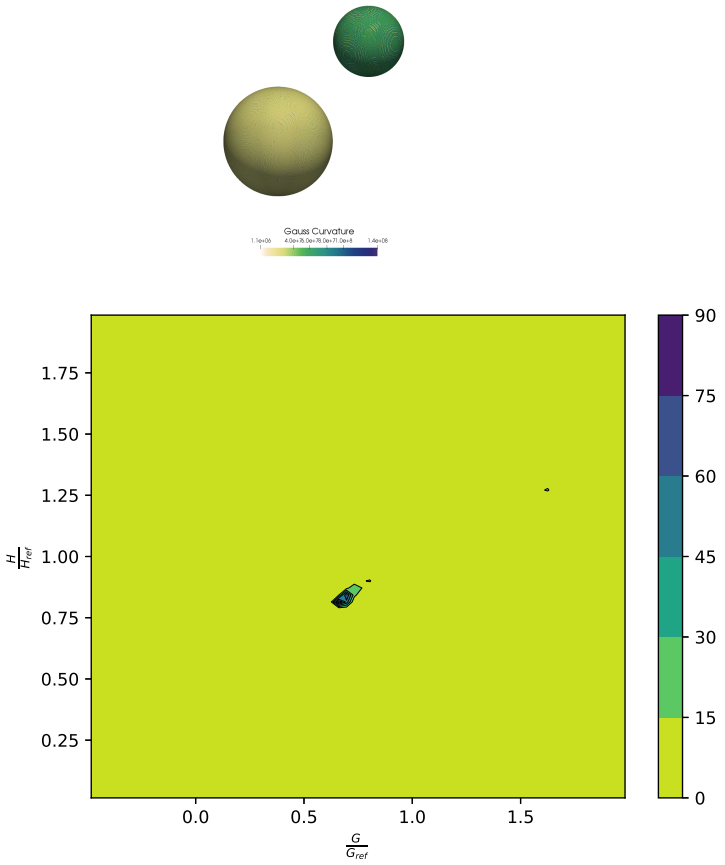


Figure 11: Beginning of the simulation ($\tau = 0$). The color mapping for the 3D objects is on the Gauss curvature G

topology of the coalesced object strongly changes, and this is reflected on the phase space with a cloud of points all close along the curve $H = \sqrt{G}$. The main mesh resolution effect is the reduction of the noise, *i.e.* points tend to stay in the physical part of the phase space ($H > \sqrt{G}$) as shown also in fig. 10 and close along the limit curve.

4 Conclusions

In this work we presented a framework to exploit geometrical information in order to gain some insight on two-phase flow behavior through the post-processing of DNS. We have shown how the algorithm of Meyer et al. (2003) fails to ensure $\Delta \geq 0$ for objects that are close in slope to a sphere and in addition it cannot be used for cell sizes that are too small since float arithmetic problems arise. We performed the analysis on a real DNS simulation at three different resolutions extracting topological maps that emphasize the different physical behaviour associated to a different surface tension estimation, or in general different physical reproducibility due to the different resolution of the computational meshes. In Essadki et al. (2017) an averaging kernel that preserves Gauss-Bonnet theorem is presented: studying the in-

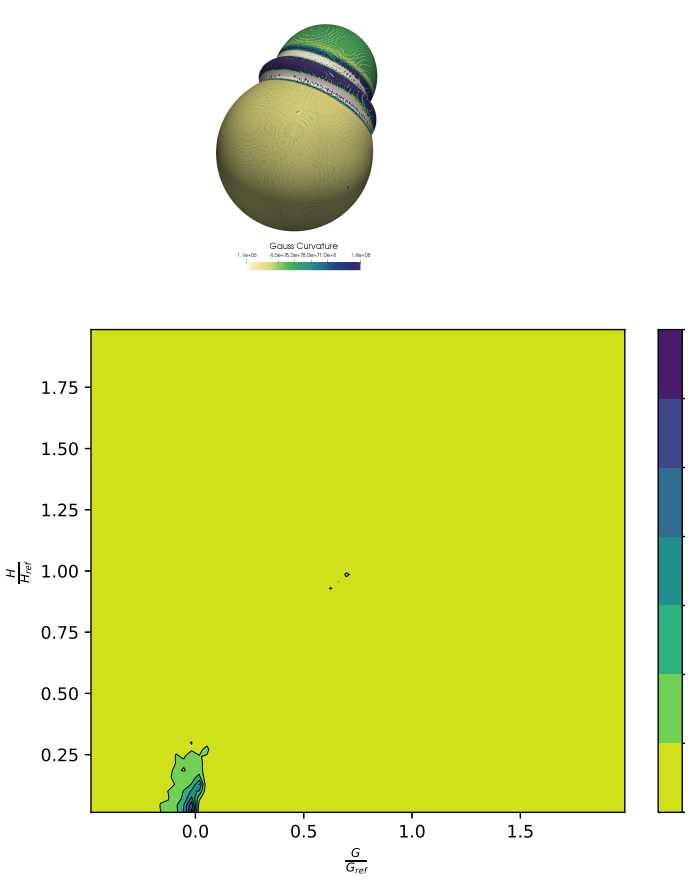


Figure 12: ($\tau = 0.3 \times 10^{-3}$). The color mapping for the 3D objects is on the Gauss curvature G

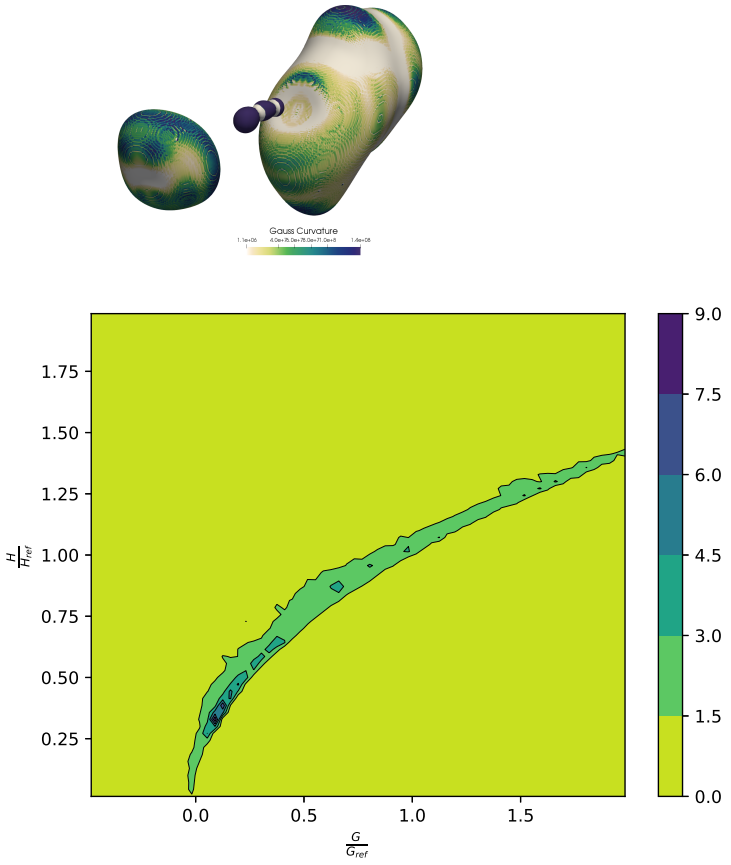
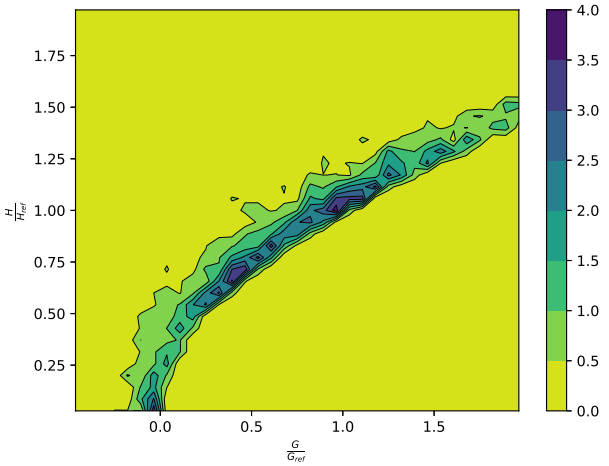
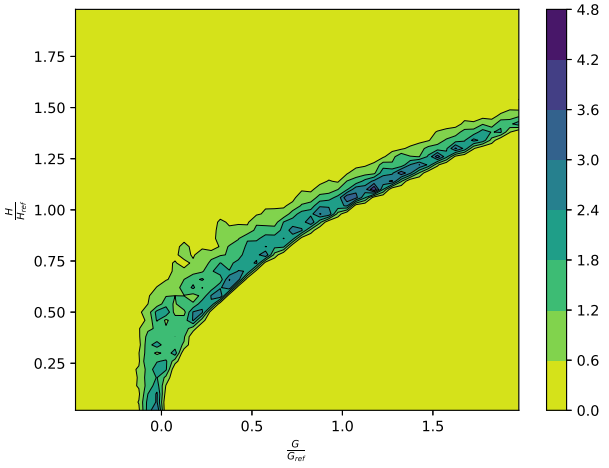


Figure 13: End of the simulation ($\tau = 1.2 \times 10^{-3}$). The color mapping for the 3D objects is on the Gauss curvature G



(a)



(b)

Figure 14: Comparison of coarse (14a) and medium resolution (14b) PDFs for $\tau = 1.2 \times 10^{-3}$

fluence of that averaging process on the outliers and phase-space statistics would be very interesting in order to achieve a more robust strategy *w.r.t.* curvature changes during topological changes, since the average would take care of bounding the maximum values of curvature in these pathological cases. A preliminary implementation in Python was already present in the work of Essadki et al., it was used to compute NDFs on DNS cases exploiting the Gauss-Bonnet preserving feature together with the averaging process that mitigates the effect of small or misshaped triangles resulting from the iso-contouring MC procedure. We plan to implement the same averaging kernel in `Mercur(v)e` as an indispensable tool to perform a complete analysis of injection cases (Blanchard (2014) and Vaudor et al. (2017)) in terms of topological maps and number statistics in order to gather enough insight to develop closure terms for the reduced models available nowadays. As an additional future perspective, comparing other algorithm to estimate curvatures on discrete surface would allow to choose the most suitable strategy for the specific interfaces that are encountered in injection systems, in terms of robustness against topological changes and performance.

References

Baer, M. R. and J. W. Nunziato (1986). "A Two-Phase Mixture Theory for the Deflagration-to-Detonation Transition (DDT) in Reactive Granular Materials". In: *International journal of multiphase flow* 12.6, pp. 861–889.

Bektas, S. (Feb. 2017). "Curvature of the Ellipsoid with Cartesian Coordinates". In: *Landscape Architecture and Regional Planning*, p. 6.

Bermejo-Moreno, I. and D. I. Pullin (May 2008). "On the Non-Local Geometry of Turbulence". In: *J. Fluid Mech* 603.

Bermejo-Moreno, I., D. I. Pullin, and K. Horiuti (Feb. 2009). "Geometry of Enstrophy and Dissipation, Grid Resolution Effects and Proximity Issues in Turbulence". In: *J. Fluid Mech.* 620, p. 121.

Blanchard, G. (2014). "Modélisation et simulation multi-échelles de l'atomisation d'une nappe liquide cisaillée". PhD Thesis. Université de Toulouse.

Cifuentes, L. et al. (Sept. 2018). "Analysis of Flame Curvature Evolution in a Turbulent Premixed Bluff Body Burner". In: *Physics of Fluids* 30.9, p. 095101.

Cordesse, P. et al. (July 2018). "Derivation of a Two-Phase Flow Model with Two-Scale Kinematics, Geometric Variables and Surface Tension Using Variational Calculus". In: *Proceedings of the NASA Summer Program 2018*. Nasa Technical Memorandum.

Di Battista, R. (2018). *Mercur(v)e — A Library to Exploit Geometrical and Topological Properties to Allow Post-Processing of DNS Simulations (and Much More)*. https://gitlab.labos.polytechnique.fr/massot-team/hgve.

Dopazo, C. et al. (July 2018). "Strain, Rotation and Curvature of Non-Material Propagating Iso-Scalar Surfaces in Homogeneous Turbulence". In: *Flow, Turbulence and Combustion* 101.1, pp. 1–32.

Drew, D. A. (1990). "Evolution of Geometric Statistics". In: *SIAM Journal on Applied Mathematics* 50.3, pp. 649–666.

Drui, F. et al. (July 2016). "A Hierarchy of Simple Hyperbolic Two-Fluid Models for Bubbly Flows". https://hal.archives-ouvertes.fr/hal-01349441.

Essadki, M. et al. (Oct. 2017). "Statistical Modeling of the Gas-Liquid Interface Using Geometrical Variables: Toward a Unified Description of the Disperse and Separated Phase Flows". (In revision for Int. J. Multiph. Flow). https://hal.archives-ouvertes.fr/hal-01615076.

Gavrilyuk, S. and H. Gouin (1999). "A New Form of Governing Equations of Fluids Arising from Hamilton's Principle". In: *International Journal of Engineering Science* 37.12, pp. 1495–1520.

Ishii, M. and K. Mishima (1984). "Two-Fluid Model and Hydrodynamic Constitutive Relations". In: *Nuclear Engineering and design* 82.2-3, pp. 107–126.

Leung, T., N. Swaminathan, and P. A. Davidson (Nov. 2012). "Geometry and Interaction of Structures in Homogeneous Isotropic Turbulence". In: *Journal of Fluid Mechanics* 710, pp. 453–481.

Lewiner, T. et al. (Jan. 2003). "Efficient Implementation of Marching Cubes' Cases with Topological Guarantees". In: *Journal of Graphics Tools* 8.2, pp. 1–15.

Ménard, T., S. Tanguy, and A. Berlemont (2007). "Coupling Level Set/VOF/Ghost Fluid Methods: Validation and Application to 3D Simulation of the Primary Break-up of a Liquid Jet". In: *Intern. J. Multiph. Flow* 33.5, pp. 510–524.

Meyer, M. et al. (2003). "Discrete Differential-Geometry Operators for Triangulated 2-Manifolds". In: *Visualization and Mathematics III*. Berlin, Heidelberg, pp. 35–57.

Pope, S. B. (1988). "The Evolution of Surfaces in Turbulence". In: *International journal of engineering science* 26.5, pp. 445–469.

incollection
Schroeder, W., K. Martin, and B. Lorensen (2006). *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. Kitware. 512 pp.

Schroeder, W., R. Maynard, and B. Geveci (Oct. 2015). "Flying Edges: A High-Performance Scalable Isocontouring Algorithm". In: IEEE, pp. 33–40. http://ieeexplore.ieee.org/document/7348069/.

Vaudor, G. et al. (2017). "A Consistent Mass and Momentum Flux Computation Method for Two Phase Flows. Application to Atomization Process". In: *Comput. Fluids* 152, pp. 204–216.