REVIEW ARTICLE

MLS-SPH-ALE: A Review of Meshless-FV Methods and a Unifying Formulation for Particle Discretizations

Antonio Eirís[1] · Luis Ramírez[2] · Iván Couceiro[2] · Javier Fernández-Fidalgo[2,3] · José París[2] · Xesús Nogueira[2]

Abstract

Mesh-based and particle methods were conceived as two different discretization strategies to solve partial differential equations. In the last two decades computational methods have diversified and a myriad of hybrid formulations that combine elements of these two approaches have been developed to solve Computational fluid dynamics problems. In this work we present a review about the meshless-FV family of methods, an analysis is carried out showing that the MLS-SPH-ALE method can be considered as a general formulation from which a set of particle-based methods can be recovered. Moreover, we show the relations between the MLS-SPH-ALE method and the finite volume method. The MLS-SPH-ALE method is a versatile particle-based method that was developed to circumvent the consistency issues of particle methods caused by the use of the kernel approximation. The MLS-SPH-ALE method is developed from the differential equation in ALE form using the partition unity property which is automatically fulfilled by the Moving Least Squares approximation.

1 Introduction

In the last decades, an extensive list of meshless formulations was published in the literature in order to circumvent some deficiencies or improve the performance of the original Smoothed Particle Hydrodynamics (SPH) formulation. Several novel techniques have been developed for simulating fluid dynamics, including Lagrangian particle-based methods that leverage kernel approximation. However, alternative approaches formulated in an Eulerian frame that utilize other approximation techniques have also emerged. As the first and the most widely used meshless method, there is some

tendency in categorize as SPH any particle based method that adopts a Lagrangian description. However, this is not the only possible choice. In this work, instead of organizing methods adopting the traditional SPH method as a reference, we prefer to define three groups of numerical methods based on a more general concept. Following the ideas of Hopkins [1], we organize numerical methods for Computational Fluid Dynamics based on the way in which the partition of the domain is accomplished. Hopkins [1] considers three different forms of subdividing the domain resulting in three groups of methods that are referred as mesh-based methods, meshless Finite Volume (meshless-FV) methods and SPH methods. For each group, the flow representation (Lagrangian, Eulerian, or Arbitrary Lagrangian–Eulerian (ALE))

Luis Ramírez, Iván Couceiro, Javier Fernández-Fidalgo, José París and Xesús Nogueira have contributed equally to this work.

✉ Luis Ramírez
luis.ramirez@udc.es

Antonio Eirís
eiris@cud.uvigo.es

Iván Couceiro
ivan.couceiro.aguiar@udc.es

Javier Fernández-Fidalgo
j.fernandez.fidalgo@upm.es

José París
jose.paris@udc.es

Xesús Nogueira
xesus.nogueira@udc.es

[1] Defense University Center at the Spanish Naval Academy, Plaza de España, s/n, 36920 Marín, Spain

[2] Group of Numerical Methods in Engineering-GMNI, Center for Technological Innovation in Construction and Civil Engineering-CITEEC, Universidade da Coruña, Civil Engineering School, Campus de Elviña, 15071 A Coruña, Spain

[3] Department of Civil Engineering: Hydraulics, Energy and Environment, Universidad Politécnica de Madrid, Calle del Profesor Aranguren 3, 28040 Madrid, Spain

and equation-solving formulation (compressible or incompressible) have been disregarded. Other classifications can be found in [2] and [3], among many others.

The classification provided by Hopkins considers an intermediate group that bridges the gap between mesh-based and SPH methods. This point of view can help to understand the different techniques and methods introduced to improve the original SPH method. For instance, particle disordering techniques used in Lagrangian SPH methods [4, 5] have some similarities with the use of the ALE framework to avoid degenerated cells in mesh-based methods. By using a meshless-FV method, Oger et al. [6] showed that particle disordering/shifting techniques need to be implemented within an ALE framework in order to avoid interpolation errors. Based on that findings, Antuono et al. [7] incorporated the ALE framework into the δ-SPH, that was conceived originally as a pure Lagrangian SPH method. Another important connection between different groups of methods is related with the stabilization of the scheme. For instance, the stabilization terms in δ-SPH can be interpreted as diffusive terms of Riemann solvers [8]. As an example of the transfer of techniques from methods of the third group to the first group we cite the use of scatter data approximation techniques that have been used in Finite Volume Methods to extend the order of reconstructions in high order Godunov methods [9, 10].

In the following, we review the main features of the different groups of Hopkins' classification, with special attention to the Group II. The rationale behind this is to introduce a comprehensive formulation for this category that can be regarded as a universal formulation, capable of generating a multitude of methods from the other groups. We believe that this general perspective can help the research community to find new more accurate and more efficient numerical schemes.

1.1 Group I: Mesh-Based Methods

In the first group, Hopkins included mesh-based methods like the Finite Volume Method (FVM), the Finite Element Method (FEM), the Finite Difference Method (FDM) and Particle Methods (among others) that use a Voronoi tessellation. In mesh-based methods the subdomain partition is the one that corresponds with the mesh topology. In Voronoi methods the information is stored in a cloud of points. A tessellation algorithm is run to associate each particle with a subdomain whose geometry is clearly defined, and there is no overlapped zone between particle subdomains. For example, in cell-centered finite volume methods (FVMs) the particle location is the barycenter of the cell meanwhile in the Voronoi tessellation the particle location is not coincident with the barycenter. In this group of methods, each particle has a clearly defined volume and area of the "particle boundary" that can be calculated with geometric formulae. Clear exposition of the FDM, FVM and FEM methods is given in classical Computational Fluid Dynamics textbooks [11–13]. For a further insight into Voronoi Tessellation Methods we recommend the publications associated with the AREPO project [14, 15].

1.2 Group II: Meshless-FV Methods

In the second group of methods, Hopkins included the particle methods in which the subdomain associated with each particle is calculated by means of functions that involve information contained in the neighborhood of the particle. Numerical methods in this group are meshless methods that share some characteristics with the FVM. It is important to note that for the methods belonging to this group, the subdomain associated with each particle is not a clearly defined geometric entity. We can compute the volume associated with a particle, but it is obtained by integration of a function which is not associated to the geometry of a cell. To remark this feature, we use the term effective volume instead of the geometric volume used by mesh-based methods. The same rationale is used to define the interaction area associated with two neighboring particles. Most of the methods in this category are derived from the Euler's equations expressed in conservative form. Among all the methods that belongs to this group, we can cite the following (non-exhaustive) list:

- Meshless Finite Volume Method (MFVM) [16]
- SPH-ALE [17–20]
- Renormalized SPH-ALE method [21, 22]
- Finite Volume Particle Method (FVPM) [23]
- Meshless Finite Volume (Gizmo-MFV) [1, 24][1]
- MLS-SPH-ALE [25, 26]
- Moving Least Squares SPH (MLSPH) [27, 28]

This review will focus on the MLS-SPH-ALE method [25, 26]. It was developed to circumvent the consistency issues of particle methods caused by the use of the kernel approximation. In this work we demonstrate that it can be perceived as a comprehensive formulation, from which it is possible to generate various formulations of methods belonging to the different groups of the proposed classification.

---

[1] The MFV developed by Hopkins was implemented in the Gizmo project, see http://www.tapir.caltech.edu/ phopkins/Site/GIZMO.html. The Gizmo-MFV has also great similarities with the MFV proposed by Gaburov and Nitadori.

Table 1 Classification of the different numerical methods for Computational Fluid Dynamics following [1]

Group	Method	References
I: Mesh-based Methods	Finite Difference Method (FDM)	[11–13]
	Finite Volume Method (FVM)	[11–13]
	Finite Element Method (FEM)	[11–13]
II: Meshless-FV methods	Meshless Finite Volume Method (MFVM)	[16]
	SPH-ALE	[17–20]
	Renormalized SPH-ALE method	[21, 22]
	Finite Volume Particle Method (FVPM)	[23]
	Meshless Finite Volume (Gizmo-MFV)	[1, 24]
	MLS-SPH-ALE	[25, 26]
	Moving Least Squares SPH (MLSPH)	[27, 28]
III: SPH methods	Traditional SPH	[29, 30]
	δ-SPH	[31]
	Riemann-SPH	[32, 33]
	Finite Point Method	[34]
	Finite Pointset Method	[35, 36]

Underlined methods are derived in this paper from the MLS-SPH-ALE method

1.3 Group III: SPH Methods

The third group includes particle methods that approximate the continuum without requiring a geometric subdivision of the domain. The most representative method of this group is the SPH method [29, 30], which is a Lagrangian method that assigns a fixed mass to each particle. As the numerical model updates the solution, particle density evolves and it is used to compute the volume associated to the particle. We note that this is a fictitious volume because SPH methods only require mass-based weights. In this group we include the original SPH method and some other formulations, like the δ-SPH [31] and Riemann-SPH [32, 33] methods. Discretized equations for methods in this group are derived by replacing the continuous operators in the differential equations by discrete operators expressed in terms of the kernel function. We also include in this group other particle methods like the Finite Point Method [34] and the Finite Pointset Method. Note that the use of the kernel approximation is not required to belong to this group. Indeed, Finite Point and Finite Poinset Methods [35, 36] do not use the kernel approximation, and the cloud of points does not require mass-based weights. Among all the methods of this group, we can cite the following (non-exhaustive) list:

- Traditional SPH [29, 30]
- δ-SPH [31]
- Riemann-SPH [32, 33]
- Finite Point Method [34]
- Finite Pointset Method [35, 36]

1.4 Comparisons Between Methods

In the literature, comparisons between methods are usually carried out by conducting benchmarking studies over test cases [37]. In this work, we follow a different approach. We focus on the derivation process to highlight the existing connections among the different formulations. Note that a similar strategy was adopted in the study conducted by Schaller et al. [38]. In our analysis, we consider the MLS-SPH-ALE method as a general meshless formulation. Thus, the MLS-SPH-ALE method is derived starting from the differential equations set. Once the semi-discrete equations are obtained we show that other formulations can be derived by introducing some particular settings in the MLS-SPH-ALE formulation. In Table 1, we summarize the classification of the different methods and we highlight the methods that we will derive from the MLS-SPH-ALE method.

2 Governing Equations

2.1 Eulerian Form

Governing equations in Eulerian form are obtained by imposing the balance equations to a fixed region in space. The conservative form of Navier–Stokes equations is given by

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}_E^\alpha}{\partial x^\alpha} = \frac{\partial \boldsymbol{D}^\alpha}{\partial x^\alpha} + \boldsymbol{S} \tag{1}$$

where $\boldsymbol{U}$ is the vector of conservative variables, $\boldsymbol{F}_E^\alpha$ is the Eulerian flux in direction α of the advective and pressure terms, $\boldsymbol{D}^\alpha$ is the viscous flux in direction α and $\boldsymbol{S}$ is the vector of source terms.

$$U = \begin{pmatrix} \rho \\ \rho u^\beta \\ \rho E \end{pmatrix}, \quad F_E^\alpha = \begin{pmatrix} \rho u^\alpha \\ \rho u^\beta u^\alpha + p\delta^{\beta\alpha} \\ \rho H u^\alpha \end{pmatrix}$$

$$D^\alpha = \begin{pmatrix} 0 \\ \tau^{\alpha\beta} \\ k\frac{\partial T}{\partial x^\alpha} + \tau^{\alpha\beta} u^\beta \end{pmatrix}, \quad S = \begin{pmatrix} 0 \\ \rho f^\beta \\ \rho f^\beta u^\beta + \dot{q}_h \end{pmatrix}$$

where Greek superscripts with summation convention are used to refer spatial directions in a Cartesian Coordinate System with d dimensions. Greek superscripts α, β range from 1 to d and are associated with the Cartesian directions in the usual form $1 \equiv x, 2 \equiv y, 3 \equiv z$. The fluid velocity vector is given by $\boldsymbol{u} = (u^1, \ldots, u^d)^T$. Density and pressure are designed by ρ and p respectively. Total energy is denoted by E and its relation with the internal energy e and the kinetic energy reads $E = e + \frac{1}{2}(u^{\alpha\alpha})$. Total enthalpy definition is used to identify $H = E + p/\rho$. The external force component per mass unit in momentum equation are expressed by f^α and $\dot{q}_h$ represents a volumetric heat source. For the diffusive terms, $\tau^{\alpha\beta}$ denotes the viscous tensor component, and the thermal conduction flux component is given in terms of the thermal conductivity k and the temperature gradient as $q^\alpha = -k\frac{\partial T}{\partial x^\alpha}$. Assuming a Newtonian fluid, the viscous tensor is defined as

$$\tau^{\alpha\beta} = \mu \left(\frac{\partial u^\beta}{\partial x^\alpha} + \frac{\partial u^\alpha}{\partial x^\beta} \right) + \left(\mu_v - \frac{2}{3}\mu \right)(\nabla \cdot \boldsymbol{u})\,\delta^{\alpha\beta}$$

where μ and μ_v are the dynamic and volumetric viscosity of the fluid respectively.

2.2 Arbitrary Lagrangian–Eulerian (ALE) Form

In order to provide compact expressions for the ALE framework and following the usual notation in literature, the transport operator is introduced here. Let's denote with $L_w(\boldsymbol{U})$ the transport operator with a regular velocity field $\boldsymbol{w}$ that acts on a vector of variables $\boldsymbol{U}$ according to

$$L_w(\boldsymbol{U}) \equiv \frac{\partial}{\partial t}\boldsymbol{U} + \nabla \cdot (\boldsymbol{w} \otimes \boldsymbol{U})$$

In a Cartesian Coordinate System with d dimensions the transport operator reads as

$$L_w(\boldsymbol{U}) = \frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial}{\partial x^\alpha}(w^\alpha \boldsymbol{U}) \tag{2}$$

The transport operator $L_w(\boldsymbol{U})$ is a differential operator whose physical interpretation is linked to the Generalized Reynolds Transport Theorem, applied to a arbitrary control volume whose boundary is moving with velocity $\boldsymbol{w}$.

Now let us add and subtract the term $\frac{\partial(w^\alpha \boldsymbol{U})}{\partial x^\alpha}$ on the left hand side of Eq. (1). We write

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial(w^\alpha \boldsymbol{U})}{\partial x^\alpha} + \frac{\partial F_E^\alpha}{\partial x^\alpha} - \frac{\partial(w^\alpha \boldsymbol{U})}{\partial x^\alpha} = \frac{\partial D^\alpha}{\partial x^\alpha} + \boldsymbol{S} \tag{3}$$

Using the transport operator definition given in Eq. (2), Eq. (3) becomes

$$L_w(\boldsymbol{U}) + \frac{\partial(F_E^\alpha - w^\alpha \boldsymbol{U})}{\partial x^\alpha} = \frac{\partial D^\alpha}{\partial x^\alpha} + \boldsymbol{S} \tag{4}$$

By introducing the ALE flux $F_w^\alpha = F_E^\alpha - w^\alpha \boldsymbol{U}$, which is linked with the transport velocity $\boldsymbol{w}$, Eq. (4) is rewritten as

$$L_w(\boldsymbol{U}) + \frac{\partial F_w^\alpha}{\partial x^\alpha} = \frac{\partial D^\alpha}{\partial x^\alpha} + \boldsymbol{S} \tag{5}$$

Equation (5) expresses in conservative vector form the Navier-Stokes equations in ALE framework. With the proper index set for the Greek superscripts, expressions can be accommodated to 1D, 2D and 3D domains. When the diffusive flux components D^α are equal to zero, the Euler equations are recovered.

3 The MLS-SPH-ALE Method

In this section, we present the formulation of the MLS-SPH-ALE method. This method belongs to Group II of Hopkins' classification, and we will show later that it can be seen as a general formulation for many schemes of the different groups. In order to develop the formulation for the Navier–Stokes equations it is convenient to group the ALE flux tensor $\mathbf{F}_w$ and the diffuse tensor $\mathbf{D}$ in total flux tensor given $\mathbf{F} \equiv \mathbf{F}_w - \mathbf{D}$. Then, Eq. (5) is rewritten as

$$L_w(\boldsymbol{U}) + \nabla \cdot \mathbf{F} = \boldsymbol{S} \tag{6}$$

We start the derivation process by multiplying Eq. (6) by the Moving Least Squares (MLS) [27, 28, 39] shape function $N_i = N(\boldsymbol{x} - \boldsymbol{x}_i, h_i)$ centered at particle i. Then, we take the integral over the domain Ω

$$\int_\Omega N_i L_w(\boldsymbol{U}) dx^d + \int_\Omega N_i \nabla \cdot \mathbf{F}\, dx^d = \int_\Omega N_i \boldsymbol{S}\, dx^d \tag{7}$$

Using the MLS approximation for the divergence of the flux and one point quadrature rule Eq. (7) gives

$$V_i L_w(\boldsymbol{U}_i) + \int_\Omega N_i \sum_{j=1}^{n_i} (\mathbf{F})_j \cdot \nabla N_j\, dx^d = V_i \boldsymbol{S}_i$$

Invoking the Generalized Reynolds Transport Theorem, the term $V_i L_w(U_i)$ can be expressed as a time derivative following the particle, as

$$\frac{d(V_i U_i)}{dt} + \int_\Omega N_i \sum_{j=1}^{n_i} (\mathbf{F})_j \cdot \nabla N_j \, dx^d = V_i S_i$$

Assuming that $(\mathbf{F})_j$ are point values, we can write

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} (\mathbf{F})_j \int_\Omega N_i \cdot \nabla N_j \, dx^d = V_i S_i \tag{8}$$

Integrating by parts and applying the divergence theorem, we obtain

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} (\mathbf{F})_j \left(\int_{\partial\Omega} N_i N_j \cdot \mathbf{n} \, dx^{d-1} \right.$$
$$\left. - \int_\Omega N_j \cdot \nabla N_i \, dx^d \right) = V_i S_i \tag{9}$$

where $\mathbf{n}$ is the unitary surface normal and $\partial\Omega$ is the boundary of the domain Ω.

If we average Eqs. (8) and (9), we obtain

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} \frac{1}{2} (\mathbf{F})_j \left(\int_{\partial\Omega} N_i N_j \cdot \mathbf{n} \, dx^{d-1} \right.$$
$$\left. - \int_\Omega N_j \cdot \nabla N_i \, dx^d + \int_\Omega N_i \cdot \nabla N_j \, dx^d \right) = V_i S_i \tag{10}$$

Now, we add $\frac{1}{2}\mathbf{F}_i$ and subtract $\mathbf{F}_i$ to the fluxes in Eq. (10) yielding to

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} \left[\frac{1}{2}(\mathbf{F}_j + \mathbf{F}_i) - \mathbf{F}_i \right]$$
$$\left(\int_{\partial\Omega} N_i N_j \cdot \mathbf{n} \, dx^{d-1} - \int_\Omega N_j \cdot \nabla N_i \, dx^d \right.$$
$$\left. + \int_\Omega N_i \cdot \nabla N_j \, dx^d \right) = V_i S_i \tag{11}$$

Note that the operations involved in the transition from Eq. (10) to Eq. (11) have no effect at the continuum level, since, using Eq. (A3) (see appendix 1) it can be proved that

$$\sum_{j=1}^{n_i} (\mathbf{F})_i \int_\Omega N_i \nabla N_j \, dx^d$$
$$= (\mathbf{F})_i \int_\Omega N_i \sum_{j=1}^{n_i} \nabla N_j \, dx^d = \mathbf{0} \tag{12}$$

Using a one-point quadrature for the volume integrals we arrive to

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} \left[\frac{1}{2}(\mathbf{F}_j + \mathbf{F}_i) - \mathbf{F}_i \right]$$
$$\left(\int_{\partial\Omega} N_i N_j \cdot \mathbf{n} \, dx^{d-1} - V_j \nabla N_{ij} + V_i \nabla N_{ji} \right) = V_i S_i \tag{13}$$

where $N_{ij} = N(\mathbf{x}_j - \mathbf{x}_i, h_i)$ and $N_{ji} = N(\mathbf{x}_i - \mathbf{x}_j, h_j)$.

Note that the semidiscrete-system described by Eq. (13) contains a boundary term that allows us to impose boundary conditions without requiring any external technique. In the MLS-SPH-ALE scheme, the boundary conditions are imposed weakly through the fluxes Eq. (13) in a similar way to FVMs. This makes the implementation of boundary conditions much easier than in other meshless approaches as SPH.

Another source of difficulty for the implementation of boundary conditions in kernel-based meshless methods is the truncation of the kernel near the boundaries. This causes large errors in the kernel approximation since the partition of unity is not fulfilled. A usual solution to avoid this issue is to add additional layers of ghost particles. However, since the MLS shape functions verify the partition of unity even in situations near the boundaries (if a minimum number of neighbors is used), the MLS-SPH-ALE method does not need to add these extra layers of ghost particles.

Note also that Eqs. (11) and (13) were obtained for the discretization of the Navier–Stokes equations. For the particular case of the Euler equations the total flux tensor given by $\mathbf{F} \equiv \mathbf{F}_w - \mathbf{D}$ is coincident to the ALE flux tensor $\mathbf{F}_w$.

In order to establish some fast comparisons with standard SPH methodology let us assume there is no rigid boundary, so the boundary integral is equal to zero, and the discretization obtained by MLS-SPH-ALE for the Euler equations reads

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} \left[\frac{1}{2}\left((\mathbf{F}_w)_j + (\mathbf{F}_w)_i \right) - (\mathbf{F}_w)_i \right]$$
$$\left(-V_j \nabla N_{ij} + V_i \nabla N_{ji} \right) = V_i S_i \tag{14}$$

Finally, we define the numerical flux as $\mathbf{G}_{ij} \approx \frac{1}{2}\left((\mathbf{F}_w)_j + (\mathbf{F}_w)_i \right)$ (the difference in the two terms is the numerical dissipation).

Therefore, the MLS-SPH-ALE discretization of the system of conservation laws given by Eq. (6) reads as

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} \left[\mathbf{G}_{ij} - (\mathbf{F}_w)_i \right]$$
$$\left(-V_j \nabla N_{ij} + V_i \nabla N_{ji} \right) = V_i S_i \tag{15}$$

To complete the discretization, it is required an equation for the motion of the particles, which in this case reads as

$$\frac{d\boldsymbol{x}_i}{dt} = \sum_{j=1}^{n_i} N_{ij}\boldsymbol{w}_j \tag{16}$$

As in the traditional SPH, the equation of evolution of the particles volumes follows a similar rationale. For a generic function $f(\boldsymbol{x}, t)$ the generalization of the Leibniz rule gives

$$\frac{d}{dt}\int_\Omega f(\boldsymbol{x}, t)\, dx^d = \int_\Omega \frac{\partial f(\boldsymbol{x}, t)}{\partial t}\, dx^d + \int_{\partial\Omega} f(\boldsymbol{x}, t)\boldsymbol{w}\cdot\boldsymbol{n}\, dx^{d-1} \tag{17}$$

If we consider the constant function, $f(\boldsymbol{x}, t) = 1$, a control volume Ω equal to V_i and the velocity at the boundary equal to $\boldsymbol{w}$ we obtain

$$\frac{d}{dt}\int_{V_i} dx^d = \int_{\partial V_i}\boldsymbol{w}\cdot\boldsymbol{n}\, dx^{d-1} = \int_{V_i}\nabla\cdot\boldsymbol{w}\, dx^d \tag{18}$$

Introducing the MLS approximation,

$$\frac{d}{dt}\int_{V_i} dx^d = \sum_{j=1}^{n_i}\int_{V_i}\boldsymbol{w}_j\nabla\cdot N_{ji}\, dx^d \tag{19}$$

Following the same procedure as in the system of conservation laws we obtain

$$\frac{dV_i}{dt} = \sum_{j=1}^{n_i}\left[\boldsymbol{w}_{ij} - \boldsymbol{w}_i\right]\left(-V_j\,\nabla N_i + V_i\,\nabla N_{ji}\right) \tag{20}$$

It is important to remark that the numerical scheme is formally second-order, even though the reconstruction of Riemann states can be higher-order [1].

4 Relations Between Groups of Numerical Methods

In this section we show the connections between the different groups of the Hopkins' classification. Through the analysis of the discretized equations, we will demonstrate that it is possible to consider that both, FVM and SPH are contained into the MFVM class of methods.

4.1 Mesh-Based Discretized Equations

We derive the discretized equations of the FVM as a representative method of the mesh-based group. The conservative formulation of Euler equations with ALE description is written in integral form as
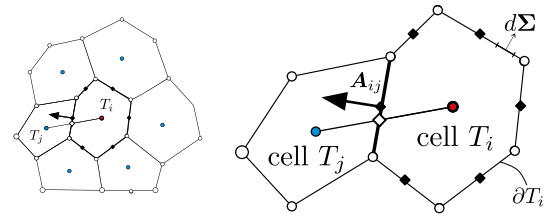

Fig. 1 Discretization of the domain by the Finite Volume Method

$$\frac{d}{dt}\int_T \boldsymbol{U}dV + \oint_{\partial T}\mathbf{F}_w\cdot d\boldsymbol{\Sigma} = \int_T \boldsymbol{S}dV \tag{21}$$

where T is the region of space associated with a cell and ∂T its boundary. $\boldsymbol{U}$ is the vector of conservative variables, $\mathbf{F}_w$ is the ALE tensor flux and $\boldsymbol{S}$ the vector of source terms.

In Fig. 1 we plot the centroids (filled points) of a group of cells. The cell boundary is given by a polygon when a bidimensional domain is considered. We note that the cell labeled with the index i has a finite number of neighboring cells n_i, that for the particular set depicted is $n_i = 6$. By focusing on one of its neighbors, labeled with the index j, an enlarged view of cells T_i and T_j is shown on the right. We designate with $\boldsymbol{A}_{ij}$ the surface shared by neighboring cells i and j. We note that $\boldsymbol{A}_{ij}$ is a vector with a module equal to the area of the shared face and the direction given by the outward normal for cell i. It can be noticed that the direction of $\boldsymbol{A}_{ij}$ is generally not parallel to the line connecting the position of the centroids. As we are using the ALE flux tensor the expression is valid for fixed, moving and deforming cells. We could also use the same notation for particle methods that employ a tessellation but in that case the full circles represent the generators and not the barycenter of the cells [40].

On an arbitrary cell T_i, the Euler equations have a simple spatial discretization given by

$$\frac{d}{dt}\left(\bar{\boldsymbol{U}}_i V_i\right) + \sum_{j\in\partial T_i}(\mathbf{F}_w\cdot\boldsymbol{A})_{ij} = \bar{\boldsymbol{S}}_i V_i \tag{22}$$

Here $\bar{\boldsymbol{U}}_i$ and $\bar{\boldsymbol{S}}_i$ are the mean values of $\boldsymbol{U}$ and $\boldsymbol{S}$ inside a mesh cell T_i which has volume V_i.[2] Note that the sum is carried over the boundaries of the cell T_i designated as ∂T_i. For convenience we introduce a compact notation for the surface integral of a single face separating cell i and j

$$(\mathbf{F}_w\cdot\boldsymbol{A})_{ij} = \int_{\partial T_{i,j}}\mathbf{F}_w\cdot d\boldsymbol{\Sigma} \tag{23}$$

This simple derivation used by FVM can be accomplished because the volume and boundaries of a cell are geometric

---

[2] For 2D domains consider that $V_i \equiv \boldsymbol{A}_i\cdot\boldsymbol{1}$.

entities associated with the mesh. Unfortunately this fast procedure cannot be followed in meshless methods, since in those methods particles represent a region of the continuous domain, but that region lacks any boundary definition to accomplish the integrals as expressed in Eq. (21). Thus, meshless methods must follow a different procedure to discretize the governing equations. Two of these procedures will be outlined in the following sections.

4.2 Meshless-FV Discretized Equations

As a representative example for meshless-FV methods we have selected the method proposed by Ivanova et al. [16]. In their work, an appendix is provided with title "On a spatially-discrete formalism for mesh-less finite-volume-methods". We use the acronym MFVM to refer to the method proposed in the appendix of their publication. The paper reviews hydrodynamic codes for cosmological applications where Lagrangian SPH and Cartesian Eulerian methods were conceived as two totally different approaches. Authors proved that Lagrangian particle-based methods are a subset of more general meshless-FV schemes. The publication encourages to take a flexible approach in the development of numerical methods and opens the path for developing hybrid codes that leverages the benefits of both types of methods.

In [16], it was presented a generic formalism that leads to spatially discrete meshless finite volume equations. However, no results were presented in the article as the formalism was not implemented in any computer framework. The procedure begins with the continuous weak form of the Euler equations in the form

$$\forall \varphi \in C_0^2(\mathbb{R}^d \times \mathbb{R}^{+,*})$$
$$\int_{\mathbb{R}^d \times \mathbb{R}^+} \left(U L_w^*(\varphi) + F_w^\alpha \frac{\partial \varphi}{\partial x^\alpha} + S\varphi \right) dx^d \, dt = 0$$

where the integral is taken over all the space-time domain. In order to ease the notation, the adjoint transport operator $L_w^*(\varphi)$ is recognized as the advective derivative in the particle velocity field w and denoted more compactly as $\dot\varphi = \partial\varphi/\partial t + w \cdot \nabla\varphi$. We also change from index notation to vector notation to write

$$\forall \varphi \in C_0^2(\mathbb{R}^d \times \mathbb{R}^{+,*})$$
$$\int_{\mathbb{R}^d \times \mathbb{R}^+} \left(U\dot\varphi + \nabla\varphi \cdot \mathbf{F}_w + S\varphi \right) dx^d \, dt = 0 \tag{24}$$

To get a particle approximation of this ALE weak problem, a set of moving particles given by paths $x_i(t)$ is considered (with index i varying from $i = 1$ to $i = N$). There are several techniques to integrate the weak form in meshless methods [41], for example, nodal integration is used in SPH methods.

Instead of accomplishing the discretization process directly by using the definition of a discrete scalar product, Ivanova et al. introduce a set of functions ψ_i linked to each particle i that verifies the Partition of Unity (PU) property. Satisfying the PU property means that for any point x in the domain Ω at any instant time t the following identity holds exactly

$$\sum_i \psi(x - x_i(t)) = 1 \quad \text{or} \quad \sum_i \psi_i(x) = 1 \tag{25}$$

By invoking the Fubini's Theorem, the weak form given by Eq. (24) is rewritten as

$$\forall \varphi \in C_0^2(\mathbb{R}^d \times \mathbb{R}^{+,*})$$
$$\int_{\mathbb{R}^+} \int_{\mathbb{R}^d} (U\dot\varphi + \nabla\varphi \cdot \mathbf{F}_w + S\varphi) \, dx^d \, dt = 0 \tag{26}$$

Leaving the time integral temporally out of the derivation, attention is set on conducting the spatial discretization of the spatial integral

$$\int_{\mathbb{R}^d} (U\dot\varphi + \nabla\varphi \cdot \mathbf{F}_w + S\varphi) \, dx^d \tag{27}$$

Using the Partition of Unity property (Eq. (25)) in Eq. (27) allows us to write

$$\int_{\mathbb{R}^d} 1(U\dot\varphi + \nabla\varphi \cdot \mathbf{F}_w + S\varphi) \, dx^d$$
$$= \int_{\mathbb{R}^d} \sum_i \psi_i(x) (U\dot\varphi + \nabla\varphi \cdot \mathbf{F}_w + S\varphi) \, dx^d \tag{28}$$

where we have decided to emphasize only the argument dependence for the particle functions $\psi_i(x)$. After expanding the products, we note that each integral demands a different approximation, and for that reason we identify each integral term with Roman numbers

$$\underbrace{\int_{\mathbb{R}^d} \sum_i \psi_i(x) U\dot\varphi \, dx^d}_{I} + \underbrace{\int_{\mathbb{R}^d} \sum_i \psi_i(x) \nabla\varphi \cdot \mathbf{F}_w \, dx^d}_{II}$$
$$+ \underbrace{\int_{\mathbb{R}^d} \sum_i \psi_i(x) S\varphi \, dx^d}_{III} \tag{29}$$

Before starting the approximation of each term, we present some properties associated with the PU. The insertion of

the PU property in the integral that gives the volume of the domain yields

$$
\begin{aligned}
V &= \int_\Omega dx^d = \int_\Omega 1 \, dx^d = \int_\Omega \sum_i \psi_i(\boldsymbol{x}) \, dx^d \\
&= \sum_i \int_\Omega \psi_i(\boldsymbol{x}) \, dx^d = \sum_i V_i
\end{aligned}
\tag{30}
$$

that provides a partition of the domain Ω, by assigning to a generic particle i the volume V_i given by

$$
V_i = \int_\Omega \psi_i(\boldsymbol{x}) \, dx^d
\tag{31}
$$

This property is of great importance. It shows that an effective volume associated with a particle i is expressed as the integral of the corresponding particle function $\psi_i(\boldsymbol{x})$ over the whole domain. Particles have a compact support associated with the kernel function, but there is overlapping between the support domain of neighboring particles. The overlapping prevents the evaluation of the integral over the domain as the sum of the integrals over the support of each particle. Equations (30) and (31) show that by inserting the PU inside the integral, it is possible to express the integral over the domain as the sum of the contributions of the particles. We also remark that Eq. (31) is obtained without using a cell or invoking a tessellation procedure. Particle functions $\psi_i(\boldsymbol{x})$ are null outside the boundary domain, and therefore the region of integration $\mathbb{R}^d$ of Eq. (29) is replaced by Ω.

Coming back to the development of Eq. (27), the term I can be approximated as

$$
\begin{aligned}
I &= \sum_i \int_\Omega \psi_i(\boldsymbol{x}) U(\boldsymbol{x}) \dot{\varphi}(\boldsymbol{x}) \, dx^d \\
&\approx \sum_i \int_\Omega \psi_i(\boldsymbol{x}) U(\boldsymbol{x}_i) \dot{\varphi}(\boldsymbol{x}_i) \, dx^d \\
&= \sum_i U(\boldsymbol{x}_i) \dot{\varphi}(\boldsymbol{x}_i) \int_\Omega \psi_i(\boldsymbol{x}) \, dx^d \\
&= \sum_i U(\boldsymbol{x}_i) \dot{\varphi}(\boldsymbol{x}_i) V_i = \sum_i U_i \dot{\varphi}_i V_i
\end{aligned}
\tag{32}
$$

where the dependence of the spatial variable is included to show clearly the one point quadrature rule used for the approximation.

For the term II, the following gradient approximation is introduced $\nabla \varphi(\boldsymbol{x}, t) \approx \widehat{\nabla \varphi}(\boldsymbol{x}) = \sum_j \nabla \psi_j(\boldsymbol{x}) \varphi(\boldsymbol{x}_j) = \sum_j \nabla \psi_j(\boldsymbol{x}) \varphi_j$ yielding

$$
\begin{aligned}
II &= \int_\Omega \sum_i \psi_i(\boldsymbol{x}) \nabla \varphi(\boldsymbol{x}) \cdot \mathbf{F}_w(\boldsymbol{x}) \, dx^d \\
&\approx \int_\Omega \sum_i \psi_i(\boldsymbol{x}) \widehat{\nabla \varphi}(\boldsymbol{x}) \cdot \mathbf{F}_w(\boldsymbol{x}) \, dx^d \\
&= \int_\Omega \sum_i \psi_i(\boldsymbol{x}) \sum_j \varphi_j \nabla \psi_j(\boldsymbol{x}) \cdot \mathbf{F}_w(\boldsymbol{x}) \, dx^d
\end{aligned}
\tag{33}
$$

At this point it is possible to take advantage of other particle function property to obtain a more convenient expression for the term II. By taking the gradient of the PU identity, it can be proved that the gradient of particle functions satisfies the Partition of Nullity (PN)

$$
\underbrace{\sum_j \psi_j(\boldsymbol{x}) = 1}_{PU} \qquad \underbrace{\sum_j \nabla \psi_j(\boldsymbol{x}) = \mathbf{0}}_{PN}
\tag{34}
$$

By using the PN property we add a term with null contribution to Eq. (33) obtaining

$$
II \approx \int_\Omega \sum_i \psi_i(\boldsymbol{x}) \sum_j \left(\varphi_j - \varphi_i \right) \nabla \psi_j(\boldsymbol{x}) \cdot \mathbf{F}_w(\boldsymbol{x}) \, dx^d
\tag{35}
$$

By expanding the terms in brackets and moving the particle functions $\psi_i(\boldsymbol{x})$ inside the summation over index j, the term II reads as

$$
\begin{aligned}
II \approx \int_\Omega \Bigg(&\sum_i \sum_j \varphi_j \psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) \cdot \mathbf{F}_w(\boldsymbol{x}) \\
&- \sum_i \sum_j \varphi_i \psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) \cdot \mathbf{F}_w(\boldsymbol{x}) \Bigg) dx^d
\end{aligned}
\tag{36}
$$

The integrand is composed by two double sums over all particles of the domain. It is possible to interchange index i and j in the first term, obtaining

$$
\begin{aligned}
II \approx \int_\Omega \Bigg(&\sum_j \sum_i \varphi_i \psi_j(\boldsymbol{x}) \nabla \psi_i(\boldsymbol{x}) \cdot \mathbf{F}_w(\boldsymbol{x}) \\
&- \sum_i \sum_j \varphi_i \psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) \cdot \mathbf{F}_w(\boldsymbol{x}) \Bigg) dx^d
\end{aligned}
\tag{37}
$$

By rearranging terms and using the additive property of the integral, the II term is given by

$$
\begin{aligned}
II \approx - \sum_i \sum_j \varphi_i \int_\Omega \mathbf{F}_w(\boldsymbol{x}) \cdot \Big(&\psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) \\
&- \psi_j(\boldsymbol{x}) \nabla \psi_i(\boldsymbol{x}) \Big) dx^d
\end{aligned}
\tag{38}
$$

In order to perform the integration of term *III*, we proceed in the same manner as for term *I*, to obtain

$$III = \sum_i \int_\Omega \psi_i(\boldsymbol{x}) S(\boldsymbol{x}) \varphi(\boldsymbol{x}) \, dx^d \approx \sum_i S_i \varphi_i V_i \tag{39}$$

Introducing the approximations obtained in Eqs. (32), (38) and (39) into Eq. (26) yields

$$\int_{\mathbb{R}^+} \left(\sum_i \boldsymbol{U}_i \dot{\varphi}_i V_i \right.$$
$$- \sum_i \sum_j \varphi_i \int_\Omega \mathbf{F}_w(\boldsymbol{x}) \cdot \left(\psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) \right. \tag{40}$$
$$\left. \left. - \psi_j(\boldsymbol{x}) \nabla \psi_i(\boldsymbol{x}) \right) dx^d + \sum_i S_i \varphi_i V_i \right) dt \approx \boldsymbol{0}$$

The overdot operator was used as abbreviation for the advective derivative according to $L_w^*(\varphi) = \dot{\varphi} = \partial\varphi/\partial t + \mathbf{w} \cdot \nabla\varphi$. Integrating by parts and noting that the tests functions have compact support (in space and time) the transient term in Eq. (40) can be expressed by

$$\sum_i \int_{\mathbb{R}^+} \boldsymbol{U}_i \frac{d\varphi_i}{dt} V_i \, dt$$
$$= \sum_i \int_{\mathbb{R}^+} \frac{d(\boldsymbol{U}_i \varphi_i V_i)}{dt} \, dt - \sum_i \int_{\mathbb{R}^+} \varphi_i \frac{d(\boldsymbol{U}_i V_i)}{dt} \, dt \tag{41}$$
$$= - \int_{\mathbb{R}^+} \left(\sum_i \varphi_i \frac{d(\boldsymbol{U}_i V_i)}{dt} \right) dt$$

Substitution of Eq. (41) into Eq. (40) gives

$$\int_{\mathbb{R}^+} \left(- \sum_i \varphi_i \frac{d(\boldsymbol{U}_i V_i)}{dt} \right.$$
$$- \sum_i \sum_j \varphi_i \int_\Omega \mathbf{F}_w(\boldsymbol{x}) \cdot \left(\psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) \right. \tag{42}$$
$$\left. \left. - \psi_j(\boldsymbol{x}) \nabla \psi_i(\boldsymbol{x}) \right) dx^d + \sum_i S_i \varphi_i V_i \right) dt \approx \boldsymbol{0}$$

As the integral must approach zero for any time, the integrand should approach zero

$$\sum_i \varphi_i \left(\frac{d(\boldsymbol{U}_i V_i)}{dt} \right.$$
$$+ \sum_j \int_\Omega \mathbf{F}_w(\boldsymbol{x}) \cdot \left(\psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) \right. \tag{43}$$
$$\left. \left. \psi_j(\boldsymbol{x}) \nabla \psi_i(\boldsymbol{x}) \right) dx^d - S_i V_i \right) \approx \boldsymbol{0}$$

Since this relation must hold for every test function, the following must be satisfied

$$\frac{d(\boldsymbol{U}_i V_i)}{dt} + \sum_j \int_\Omega \mathbf{F}_w(\boldsymbol{x}) \cdot \left(\psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) \right.$$
$$\left. - \psi_j(\boldsymbol{x}) \nabla \psi_i(\boldsymbol{x}) \right) dx^d \approx \boldsymbol{S}_i V_i \tag{44}$$

Now, we note that the second term takes the form of a sum of integrals involving the active particle *i* with its neighbor particles. The term collects the action of neighbor particles *j* over the active particle *i*. We focus the attention on one individual neighbor and we note that particle functions are dimensionless and the gradient of the particle functions have dimensions of $[L]^{-1}$. Moreover, particle functions are scalar and the gradient of particle functions are vectors. The integral obeys the expression of a flux crossing a surface. Based on this analogy we represent the interaction flux of particle *j* over particle *i* as $(\mathbf{F}_w \cdot \boldsymbol{A})_{ij}$

$$(\mathbf{F}_w \cdot \boldsymbol{A})_{ij}$$
$$\equiv \int_\Omega \mathbf{F}_w(\boldsymbol{x}) \cdot \left(\psi_i(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) - \psi_j(\boldsymbol{x}) \nabla \psi_i(\boldsymbol{x}) \right) dx^d \tag{45}$$

It is important to remark that the area *A* used to denote the term $(\mathbf{F}_w \cdot \boldsymbol{A})_{ij}$ in Eq. (45) is not associated to any geometric entity. Some authors use the term *effective area* to emphasize the differences with a real area associated with the face of a cell. Using this compact nomenclature Eq. (44) can be rewritten as

$$\frac{d(\boldsymbol{U}_i V_i)}{dt} + \sum_j (\mathbf{F}_w \cdot \boldsymbol{A})_{ij} \approx \boldsymbol{S}_i V_i \tag{46}$$

The meshless formulation presented does not provide a full discretization, because the tensor flux has not been discretized and the effective area $\boldsymbol{A}_{ij}$ requires a quadrature rule to obtain a discrete approximation. Moreover, the set of particle functions satisfying the PU and PN are key in the process, but in [16] nothing is said about its definition. At this point, the discretization is not applicable in practice since it is incomplete. However, it has the advantage of offering a meshless formulation that allows easy comparison with the FVM and traditional SPH method. In Sect. 5 we introduce the MLS-SPH-ALE method, which is also included in the Meshless-FV group. We will prove that it can be seen as a general meshless formulation, since starting from it, we can derive many different methods. In Sect. 5.1.1 we will prove that from the MLS-SPH-ALE discretization we can recover the MFVM including a complete discretization of the flux.

4.3 SPH Discretized Equations

The first thing to be clarified is that there is not a unique set of discretized equations for the Euler equations using SPH methods. In the literature different strategies to discretize the Euler equations have been explored. The simplest procedure starts with the choice of a set of Euler equations in non-conservative form, and then the continuous spatial derivatives are replaced by their discrete kernel approximations. A different procedure to derive the discretized equations adopts the Lagrangian Dynamics. Using the variational principle avoids some *ad-hoc* choices introduced by the standard procedure. Excellent reviews of the SPH procedure are given in the publications by Monaghan [30], Price [42], Rosswog [43], Springel [14] and Violeau's book [44].

As SPH is a pure Lagrangian method, the more convenient presentation for the Euler equations is the non-conservative form since material derivatives appear explicitly on the left hand side of the governing equations. For the energy equation we can use the internal energy (e) or the total energy (E) as variable.

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \boldsymbol{u} \tag{47}$$

$$\frac{d\boldsymbol{u}}{dt} = -\frac{\nabla p}{\rho} + \boldsymbol{f} \tag{48}$$

$$\frac{de}{dt} = -\frac{p}{\rho} \nabla \cdot \boldsymbol{u} \qquad \frac{dE}{dt} = -\frac{1}{\rho} \nabla \cdot (p\boldsymbol{u}) + \boldsymbol{f} \cdot \boldsymbol{u} \tag{49}$$

Equation (47), Eq. (48) and Eq. (49) express the non-conservative form for the mass, momentum and energy respectively. In the right hand side of Eq. (47) the divergence of the velocity appears meanwhile in Eq. (48) the gradient of the pressure is involved. Omitting the discretization of the energy equation we note that the main discretizations invoked for SPH are conducted by one of the two forms presented hereafter.

The traditional SPH discretization was adopted by Monaghan and it is the usual set of SPH equations used in Astrophysics. The expressions assume a constant smoothing length for the particles and the semi-discrete corresponding to Eq. (47), Eq. (48) read as

$$\frac{D\rho_i}{Dt} = \sum_j m_j (\boldsymbol{u}_i - \boldsymbol{u}_j) \nabla_i W_{ij} \tag{50}$$

$$\frac{D\boldsymbol{u}_i}{Dt} = -\sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) \nabla_i W_{ij} + \boldsymbol{f}_i \tag{51}$$

An alternative SPH discretization was proposed by Colagrossi and Landrini [31]. This alternative SPH derivation is the base for the $\delta - SPH$ method and is widely used in ocean engineering [45].

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \frac{m_j}{\rho_j} (\boldsymbol{u}_i - \boldsymbol{u}_j) \nabla_i W_{ij} \tag{52}$$

$$\frac{D\boldsymbol{u}_i}{Dt} = -\sum_j m_j \left(\frac{p_i + p_j}{\rho_i \rho_j} \right) \nabla_i W_{ij} + \boldsymbol{f}_i \tag{53}$$

Both forms of the SPH equations are self-consistent because they were derived invoking the variational principle.

4.4 Connections Between FVM and MFVM

Some authors have raised the question of whether finite volume methods do really require a mesh or not [46]. In this work we try to address this question, by analyzing the discretized equations of FVM and MFVM. The similarities between the MFVM and the FVM can be clearly seen by comparison of Eqs. (22) and (46). The main difference resides in the evaluation of the fluxes given by Eqs. (23) and (45). In FVM fluxes are calculated over a face of a geometric cell, meanwhile in the MFVM fluxes are defined over an effective area expressed in terms of particle functions.

The MFVM proposed by Ivanova et al. does not provide a full discretization of the Euler equations. The set of particle functions are not defined and the flux term expressed by $(\mathbf{F}_w \cdot \boldsymbol{A})_{ij}$ needs to be numerically evaluated with the information carried by the particles. However, the lack of an approximation for the flux and the effective area allows us to appreciate more clearly the similarity of MFVM with FVM.

4.5 Connections Between MFVM and SPH

In this subsection, we outline the derivation of the traditional SPH equations and we show how to recover these equations starting from the MFVM given by Eq. (45) and Eq. (46).

Particle functions $\psi_i(\boldsymbol{x})$ satisfying the PU property play a key role in the MFVM. On the contrary, the traditional SPH equations are derived in terms of the kernel function $W_i(\boldsymbol{x})$. Although the derivations seem to be different we show the path to recover the traditional Lagrangian SPH equations starting with the MFVM. Identifying a concealed partition of unity associated with the kernel approximation is crucial for this analysis.

SPH is based on the smoothing of the mass assigned to a set of discrete points. Based on that we define the density spread in an arbitrary point $\boldsymbol{x}$ by the presence of the mass

m_i located in point $\boldsymbol{x}_i$ by $\rho_i(\boldsymbol{x}) = m_i W(\boldsymbol{x} - \boldsymbol{x}_i, h(\boldsymbol{x}))$. The density at point $\boldsymbol{x}$ is obtained by summation over the neighboring particles (n_x) inside the kernel support, which is expressed mathematically by $\rho(\boldsymbol{x}) = \sum_j^{n_x} W(\boldsymbol{x} - \boldsymbol{x}_j, h(\boldsymbol{x})) m_j$. The following partition of unity associated with SPH can be inferred

$$
\begin{aligned}
\psi_i^{SPH}(\boldsymbol{x}) \\
= \frac{m_i W(\boldsymbol{x} - \boldsymbol{x}_i, h(\boldsymbol{x}))}{\sum_j^{n_x} W(\boldsymbol{x} - \boldsymbol{x}_j, h(\boldsymbol{x})) m_j} = \frac{\rho_i(\boldsymbol{x})}{\rho(\boldsymbol{x})}
\end{aligned}
\tag{54}
$$

Before dealing with the conservative equations, it is illustrative to examine the expression of the effective volume associated with a particle in MFVM

$$
\begin{aligned}
V_i &= \int_\Omega \psi_i^{SPH}(\boldsymbol{x}) \, dx^d \\
&= \int_\Omega \frac{m_i}{\rho(\boldsymbol{x})} W(\boldsymbol{x} - \boldsymbol{x}_i, h(\boldsymbol{x})) \, dx^d
\end{aligned}
\tag{55}
$$

It is possible to proceed further after approximating the kernel by the delta function, namely $W(\boldsymbol{x} - \boldsymbol{x}_i, h(\boldsymbol{x})) \approx \delta(\boldsymbol{x} - \boldsymbol{x}_i)$. By means of the integral identity of the delta function the following relation is derived

$$
\begin{aligned}
V_i &= \int_\Omega \frac{m_i}{\rho(\boldsymbol{x})} W(\boldsymbol{x} - \boldsymbol{x}_i, h(\boldsymbol{x})) \, dx^d \\
&\approx \int_\Omega \frac{m_i}{\rho(\boldsymbol{x})} \delta(\boldsymbol{x} - \boldsymbol{x}_i) \, dx^d = \frac{m_i}{\rho_i}
\end{aligned}
\tag{56}
$$

In traditional SPH the mass of the particle is kept fixed, and the density is derived by using the kernel approximation. It is possible to obtain a volume associated with the particle through the density definition, but it is not required for solving the governing equations. On the contrary, in the MFVM the mass of the particle is evolved from the mass conservation (Eq. (46) with $U^1 = \rho$) and the particle volume is defined by Eq. (31). To obtain a numerical value of the volume we need a quadrature rule to approximate the previous integral. There is also the alternative choice of evolving the volume of the particle using the generalization of the Leibniz rule (see Eq. (20)).

For the momentum equation, we start with the general equation of the meshless finite volume formulation given by Eq. (46) and we set the particular settings corresponding to SPH. Particles move with the fluid velocity ($\boldsymbol{w} = \boldsymbol{u}$), and the vector of conservative variables $\boldsymbol{U}_i$ and source terms $\boldsymbol{S}_i$ takes the particular form $\boldsymbol{U}_i = \rho_i \boldsymbol{u}_i$ and $\boldsymbol{S}_i = \rho_i \boldsymbol{f}_i$ thus giving

$$
\frac{d}{dt}\bigg|_u \rho_i \boldsymbol{u}_i V_i + \sum_j (\mathbf{F_u} \cdot \boldsymbol{A})_{ij} = \rho_i \boldsymbol{f}_i V_i
\tag{57}
$$

We then proceed by computing the flux terms $(\mathbf{F_u} \cdot \boldsymbol{A})_{ij}$. Since SPH is a Lagrangian method, the flux of the momentum equations is a diagonal tensor $\mathbf{F_u} = p\mathbf{I}$ involving only the pressure term. With these considerations, the general expression for the meshless flux given by Eq. (45) reads as

$$
\begin{aligned}
(\mathbf{F_u} \cdot \boldsymbol{A})_{ij} \\
= \int_\Omega \left[\psi_i(\boldsymbol{x}) p(\boldsymbol{x}) \nabla \psi_j(\boldsymbol{x}) - \psi_j(\boldsymbol{x}) p(\boldsymbol{x}) \nabla \psi_i(\boldsymbol{x}) \right] dx^d
\end{aligned}
\tag{58}
$$

Now, we introduce the expressions for $\psi_i(\boldsymbol{x})$ and $\nabla \psi_i(\boldsymbol{x})$ that correspond to the SPH kernel approximation in Eq. (58), that is

$$
\begin{aligned}
\psi_i^{SPH}(\boldsymbol{x}) &= \frac{m_i}{\rho(\boldsymbol{x})} W(x - x_i, h(\boldsymbol{x})) \\
\nabla \psi_i^{SPH}(\boldsymbol{x}) &\approx \frac{m_i}{\rho(\boldsymbol{x})} \nabla W(x - x_i, h(\boldsymbol{x}))
\end{aligned}
\tag{59}
$$

resulting in the following expression for the momentum flux $(\mathbf{F_u} \cdot \boldsymbol{A})_{ij}$

$$
\begin{aligned}
(\mathbf{F_u} \cdot \boldsymbol{A})_{ij} \\
\approx \int_\Omega \Bigg[\frac{m_i}{\rho(\boldsymbol{x})} W(\boldsymbol{x} - \boldsymbol{x}_i, h(\boldsymbol{x})) p(\boldsymbol{x}) \frac{m_j}{\rho(\boldsymbol{x})} \nabla W \\
- \frac{m_j}{\rho(\boldsymbol{x})} W(\boldsymbol{x} - \boldsymbol{x}_i, h(\boldsymbol{x})) p(\boldsymbol{x}) \frac{m_i}{\rho(\boldsymbol{x})} \nabla W \Bigg] dx^d
\end{aligned}
\tag{60}
$$

where we have set $\nabla W = \nabla W(\boldsymbol{x} - \boldsymbol{x}_i, h(\boldsymbol{x}))$ for the sake of clarity. The momentum flux $(\mathbf{F_u} \cdot \boldsymbol{A})_{ij}$ given by Eq. (60) involves a complex integration including the kernel and gradient kernel for particles i and j. However, by shrinking the kernel in the delta function we can use the properties of the delta function to obtain the approximation

$$
\begin{aligned}
(\mathbf{F_u} \cdot \boldsymbol{A})_{ij} &\approx \frac{m_i m_j}{\rho_i^2} p_i \nabla_i W(\boldsymbol{x}_i - \boldsymbol{x}_j, h_i) \\
&- \frac{m_i m_j}{\rho_j^2} p_j \nabla_j W(\boldsymbol{x}_j - \boldsymbol{x}_i, h_j)
\end{aligned}
\tag{61}
$$

Replacing Eq. (61) into Eq. (57) yields

$$
\begin{aligned}
\frac{d}{dt}\bigg|_u m_i \boldsymbol{u}_i + \sum_j \Bigg[\frac{m_i m_j}{\rho_i^2} p_i \nabla_i W(x_i - x_j, h_i) \\
- \frac{m_i m_j}{\rho_j^2} p_j \nabla_j W(x_j - x_i, h_j) \Bigg] = \rho_i \boldsymbol{f}_i V_i
\end{aligned}
\tag{62}
$$

For a uniform smoothing length $h = h_i = h_j$, we could use the symmetric property of the kernel function

$$
\begin{aligned}
\nabla_i W(\boldsymbol{x}_i - \boldsymbol{x}_j, h) &= - \nabla_j W(\boldsymbol{x}_j - \boldsymbol{x}_i, h) \\
\nabla_i W_{ij} &= - \nabla_j W_{ji}
\end{aligned}
\tag{63}
$$

and after dividing by the constant mass m_i of the particle we arrive at the final expression for the momentum equation given by

$$\frac{d}{dt}\bigg|_{u} u_i = -\sum_j m_j\left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2}\right)\nabla_i W_{ij} + f_i \qquad (64)$$

The discrete momentum equation given by Eq. (64) is coincident with Eq. (51) obtained with the traditional SPH formulation by Monaghan. Thus, it can be concluded that the SPH method can be considered as a particular configuration of the MFVM.

5 MLS-SPH-ALE: A Common Framework for Meshless-FV Methods

The hybrid features of meshless-FV methods have attracted the attention of many researchers to develop meshless methods that outperform the mesh-based and SPH methods in solving a wide variety of applications. As a consequence, several meshless-FV methods were conceived in the last decade. Some of them were developed to solve astrophysical problems ([1, 24, 47, 48]).

Some other meshless-FV methods like the Finite Volume Particle Method [23] and the SPH-ALE of Vila [17] were conceived as a general method to solve hyperbolic conservation laws. Meshless-FV methods have not achieved yet the same degree of maturity as the FEM and FVM, so it seems reasonable that in the current state of development several of the meshless-FV methods cited will evolve and some new meshless formulations will appear in a near future. Thus, the list presented in the next section is nonexhaustive but representative of the different communities currently using meshless-FV methods. In the next sections we will prove that the MLS-SPH-ALE is a general meshless-FV formulation. After a selection of representative meshless-FV in the literature, it is shown that adopting some particular settings in MLS-SPH-ALE leads to equivalent semi-discrete equations.

5.1 The Meshless Finite Volume Method (MFVM)

As comented in Sect. 4.2, the semi-discrete form of the MFVM proposed by Ivanova et al. [16] is given by Eq. (46)

$$\frac{d\left(U_i V_i\right)}{dt} + \sum_j (\mathbf{F}_w \cdot A)_{ij} \approx S_i V_i$$

where the interaction term of particle i over j, denoted as $(\mathbf{F_w} \cdot A)_{ij}$, is defined by

$$(\mathbf{F}_w \cdot A)_{ij}$$
$$\equiv \int_\Omega \mathbf{F}_w(x) \cdot \left(\psi_i(x)\nabla\psi_j(x) - \psi_j(x)\nabla\psi_i(x)\right) dx^d$$

The MFVM does not provide a full discretization for the flux between particles and for the effective area because neither of them are given in terms of data contained in the set of particles. This incomplete discretization has several implications. On one hand, without additional discretization the method is unable to run simulations. On the other hand, the method has general features that are useful to appreciate approximations invoked by other meshless-FV methods.

5.1.1 MFVM and MLS-SPH-ALE

The semi-discrete system obtained with MFVM for the Euler equations is given by Eq. (46) as

$$\frac{d\left(U_i V_i\right)}{dt} + \sum_j \int_\Omega \mathbf{F}_w(x) \cdot \left(\psi_i(x)\nabla\psi_j(x)\right.$$
$$\left. - \psi_j(x)\nabla\psi_i(x)\right) dx^d \approx S_i V_i$$

The semi-discrete system obtained with MLS-SPH-ALE method for the Navier Stokes equations is given by Eq. (11). Particularized for the Euler equations ($F = F_w$) and omitting the boundary term we obtain

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i}\left[\frac{1}{2}\left((\mathbf{F}_w)_j + (\mathbf{F}_w)_i\right) - (\mathbf{F}_w)_i\right]$$
$$\left(\int_\Omega N_i \cdot \nabla N_j\, dx^d - \int_\Omega N_j \cdot \nabla N_i\, dx^d\right) = V_i S_i$$

Noting that MFVM and MLS-SPH-ALE semi-discrete forms only differ in their expressions for the term that accounts for the interaction between particles, we can identify them writing

$$\int_\Omega \mathbf{F}_w(x) \cdot \left(\psi_i(x)\nabla\psi_j(x) - \psi_j(x)\nabla\psi_i(x)\right) dx^d$$
$$\approx \left[\frac{1}{2}\left((\mathbf{F}_w)_j + (\mathbf{F}_w)_i\right) - (\mathbf{F}_w)_i\right] \qquad (65)$$
$$\times \left(\int_\Omega N_i \cdot \nabla N_j\, dx^d - \int_\Omega N_j \cdot \nabla N_i\, dx^d\right)$$

Equation (65) shows that the MLS-SPH-ALE method assumes a constant flux tensor in the overlap region, which can be taken out of the integral. Both formulations share the same form for the effective interaction area with the only exception that MFVM is given in terms of $\psi(x)$ and MLS-SPH-ALE uses $N_i(x)$ obtained with MLS. In MLS-SPH-ALE, a full discretization of the flux is provided since the tensor flux is given in terms of particle data.

5.2 The Finite Volume Particle Method (FVPM)

The Finite Volume Particle Method (FVPM) was first developed by Hietel, Steiner and Struckmeier in [23] for the discretization of a system of conservation laws in the spatial domain $\Omega = \mathbb{R}^d$. Then, Junk and Struckmeier [49] proposed a more stable discretization in a bounded domain $\Omega \subset \mathbb{R}^d$. Keck and Hietel [50] introduced a projection technique for incompressible flow. Teleaga and Struckmeier [51] extended the method to moving domains. Quinlan and coworkers have conducted a very active research work on the FVPM on the last decade. Some of their main contributions are the extension of the FVPM to viscous flow [52] and free-surface flows [53].

The FVPM was developed to be a hybrid method with the flexibility of a particle method and the conservative nature inherent to FVM. The FVPM assigns a particle function to each particle that automatically satisfies the Partition of Unity property. This particle function is defined as

$$\psi_i(\boldsymbol{x}, t) \equiv \frac{W(\boldsymbol{x} - \boldsymbol{x}_i, t)}{\sum_{j=1}^{n_i} W(\boldsymbol{x} - \boldsymbol{x}_j, t)} = \frac{W_i(\boldsymbol{x}, t)}{\sigma(\boldsymbol{x}, t)} \quad (66)$$

where the summation over all domain particle functions at point $\boldsymbol{x}$ for an instant time t is designated more compactly as $\sigma(\boldsymbol{x}, t) \equiv \sum_{j=1}^{n_i} W(\boldsymbol{x} - \boldsymbol{x}_j, t)$. Some algorithms build the particle shape functions using kernel functions with either circular [54] or square support [55]. Quinlan et al. [56] have shown some preference for hat-shaped kernel functions over the more traditional Gaussian like kernel functions.

Note that $\boldsymbol{x}$ and t are the independent arguments of ψ_i. Therefore, the time derivative of the particle function can be inferred by using the concept of the derivative following a particle as

$$\frac{d}{dt}\psi_i(\boldsymbol{x}, t) = \frac{\partial}{\partial t}\psi_i(\boldsymbol{x}, t) + \boldsymbol{w} \cdot \nabla \psi_i(\boldsymbol{x}, t) \quad (67)$$

The expression for the derivative of the particle function is a key element in the procedure to obtain the semi-discrete equations of the FVPM. At this point it is important to remark some differences between the tasks conducted by $\psi_i(\boldsymbol{x})$ in the Gizmo-MFV method and in the FVPM. Both particle functions are used to conduct the subdivision of the domain, but in the FVPM, the $\psi_i(\boldsymbol{x}, t)$ functions are also used as test functions to define the weak form. On the other hand, in the Gizmo-MFV the particle functions $\psi_i(\boldsymbol{x})$ were introduced to facilitate the integration of the weak form. The FVPM is aimed to solve general conservation laws, written in the form

$$\frac{\partial}{\partial t}\boldsymbol{U} + \nabla \cdot \mathbf{F}_{\mathrm{E}}(\boldsymbol{U}) = \boldsymbol{S} \quad \forall \boldsymbol{x} \in \Omega(t) \subset \mathbb{R}^d, \quad t > 0 \quad (68)$$

with initial conditions $\boldsymbol{U}(\boldsymbol{x}, 0) = \boldsymbol{U}_0(\boldsymbol{x}), \forall \boldsymbol{x} \in \Omega(t = 0)$ and with suitable boundary conditions. In Eq. (68), $\Omega(t) \subset \mathbb{R}^d$ is a bounded domain in $\mathbb{R}^d$, $\boldsymbol{U}(\boldsymbol{x}, t) \in \mathbb{R}^m$, $m > 0$ is the vector of conservative variables, and $\mathbf{F}_{\mathrm{E}}(\boldsymbol{U}(\boldsymbol{x}, t))$ is the Eulerian Flux tensor.

The procedure to derive the semi-discrete equations of FVPM, starts by testing the conservation law given by Eq. (68) against the particle functions $\psi_i(\boldsymbol{x}, t)$ given by Eq. (66), resulting in the weak form given by

$$\int_{\Omega(t)} \left(\frac{\partial}{\partial t}\boldsymbol{U} + \nabla \cdot \mathbf{F}_{\mathrm{E}}(\boldsymbol{U}) \right) \psi_i(\boldsymbol{x}, t) \, dx^d$$
$$= \int_{\Omega(t)} \boldsymbol{S} \, \psi_i(\boldsymbol{x}, t) \, dx^d, \quad i = 1, \dots, N \quad (69)$$

After performing integration by parts and some convenient arrangement of terms Eq. (69) leads to

$$\frac{d(\boldsymbol{U}_i V_i)}{dt} = \sum_{j=1}^{N} \int_{\Omega(t)} \left[\left(\mathbf{F}_{\mathrm{E}} - \boldsymbol{U} \otimes \boldsymbol{w}_i(t) \right) \boldsymbol{\Gamma}_{ji}(\boldsymbol{x}, t) \right.$$
$$\left. - \left(\mathbf{F}_{\mathrm{E}} - \boldsymbol{U} \otimes \boldsymbol{w}_j(t) \right) \boldsymbol{\Gamma}_{ij}(\boldsymbol{x}, t) \right] dx^d + \boldsymbol{S}_i V_i \quad (70)$$

where $\boldsymbol{\Gamma}_{ij}(\boldsymbol{x}, t)$ and $\boldsymbol{\Gamma}_{ji}(\boldsymbol{x}, t)$ are vector functions localized on the intersection of the supports of particle i and particle j, and are defined by

$$\boldsymbol{\Gamma}_{ij}(\boldsymbol{x}, t) \equiv \psi_i(\boldsymbol{x}, t)\frac{\nabla W_j(\boldsymbol{x}, t)}{\sigma(\boldsymbol{x}, t)} \quad (71)$$

$$\boldsymbol{\Gamma}_{ji}(\boldsymbol{x}, t) \equiv \psi_j(\boldsymbol{x}, t)\frac{\nabla W_i(\boldsymbol{x}, t)}{\sigma(\boldsymbol{x}, t)} \quad (72)$$

At this moment, it is convenient to focus on the flux expressions appearing inside the brackets of Eq. (70). To evaluate the interaction of particle i with particle j we need to evaluate the fluxes $\left(\mathbf{F}_{\mathrm{E}} - \boldsymbol{U} \otimes \boldsymbol{w}_i(t) \right)$ and $\left(\mathbf{F}_{\mathrm{E}} - \boldsymbol{U} \otimes \boldsymbol{w}_j(t) \right)$. The Eulerian flux tensor $\mathbf{F}_{\mathrm{E}}(\boldsymbol{U}) = \mathbf{F}_{\mathrm{E}}(\boldsymbol{x}, t)$ and the vector of conservative variables $\boldsymbol{U}(\boldsymbol{x}, t)$ are continuous functions of space and time. On the other hand, particle velocities $\boldsymbol{w}_i(\boldsymbol{x}_i, t)$ and $\boldsymbol{w}_{\boldsymbol{x}_j j}(t)$ are functions of the position of the particle and time. To evaluate the integral of Eq. (70), an approximation of $\boldsymbol{U}$ at the overlap of particles i and j is required.

The approximation used to proceed assumes that $\boldsymbol{U}(\boldsymbol{x}, t)$ variations are small within the overlap of particles i and j, and can be represented by a single value denoted $\overline{\boldsymbol{U}}_{ij}$. Thus, we denote by $\overline{\boldsymbol{w}}_{ij}$ to the single value transport velocity in the overlap region of particles i and j. The approximations used renders that both modified fluxes inside brackets of Eq. (70) are approximately equal and uniform in each overlap region.

Then, the semi-discrete form for the FVPM reads as

$$\frac{d(U_i V_i)}{dt} + \sum_{j=1}^{N} \mathbf{F}_w\left(\overline{U}_{ij}, \overline{w}_{ij}\right) \cdot A_{ij}(t) \approx S_i V_i \qquad (73)$$

where the effective interaction area $A_{ij}(t)$ between particle i and j is given by

$$A_{ij}(t) = \int_{\Omega(t)} \left(\psi_i(\mathbf{x}, t) \frac{\nabla W_j(\mathbf{x}, t)}{\sigma(\mathbf{x}, t)} - \psi_j(\mathbf{x}, t) \frac{\nabla W_i(\mathbf{x}, t)}{\sigma(\mathbf{x}, t)} \right) dx^d \qquad (74)$$

5.2.1 FVPM and MLS-SPH-ALE

There are differences in the derivation procedures to obtain the semi-discrete form of the FVPM and MLS-SPH-ALE. The derivation of the FVPM was started from the strong form of the conservation law in Eulerian form (Eq. (68)) meanwhile the MLS-SPH-ALE started from the conservation law in ALE form (Eq. (6)). The FVPM and the MLS-SPH-ALE also introduce their approximations at different stages in the procedure. FVPM uses particle functions $\psi_i(\mathbf{x}, t)$ meanwhile MLS-SPH-ALE uses particle functions based on MLS $N_i(\mathbf{x}, t)$. When the base selected for MLS comprises only the constant term, MLS shape functions are identical to the particle functions $\psi_i(\mathbf{x}, t)$ used by FVPM. Both formulations share that both set of particle functions satisfy the PU property. A difference between the methods is that FVPM defines an effective surface interaction area $A_{ij}(t)$ that can be integrated using different quadrature rules [57]. However, it is possible to obtain some relations between these two methods by analyzing their respective semi-discrete forms.

The area $A_{ij}(t)$ in Eq. (74) is given in terms of particle functions $\psi(\mathbf{x}, t)$ and kernel gradients $\nabla W(\mathbf{x}, t)$ of the interacting particles. In order to compare with MLS-SPH-ALE formulation we look for an alternative expression of Eq. (74) given in terms of particle functions $\psi(\mathbf{x}, t)$ and their gradients $\nabla \psi(\mathbf{x}, t)$. By taking the gradient of Eq. (66), which defines the particle function $\psi_i(\mathbf{x}, t)$ associated to particle i, we obtain

$$\begin{aligned} \nabla \psi_i(\mathbf{x}, t) &= \frac{\nabla W_i(\mathbf{x}, t)}{\sigma(\mathbf{x}, t)} + W_i(\mathbf{x}, t) \nabla \left(\frac{1}{\sigma(\mathbf{x}, t)} \right) \\ &= \frac{\nabla W_i(\mathbf{x}, t)}{\sigma(\mathbf{x}, t)} - \psi_i(\mathbf{x}, t) \frac{\nabla \sigma(\mathbf{x}, t)}{\sigma(\mathbf{x}, t)} \end{aligned} \qquad (75)$$

where $\nabla W_i(\mathbf{x}, t)$ can be isolated to obtain

$$\nabla W_i(\mathbf{x}, t) = \nabla \psi_i(\mathbf{x}, t) \sigma(\mathbf{x}, t) + \psi_i(\mathbf{x}, t) \nabla \sigma(\mathbf{x}, t) \qquad (76)$$

Similarly, we can obtain an equivalent expression for $\nabla W_j(\mathbf{x}, t)$

$$\nabla W_j(\mathbf{x}, t) \quad = \nabla \psi_j(\mathbf{x}, t) \sigma(\mathbf{x}, t) + \psi_j(\mathbf{x}, t) \nabla \sigma(\mathbf{x}, t) \qquad (77)$$

By inserting Eq. (76) and Eq. (77) into Eq. (74) the terms involving products $\psi_i(\mathbf{x}, t) \psi_j(\mathbf{x}, t)$ are canceled, obtaining

an expression for $A_{ij}(t)$ in terms of particle functions, that reads as

$$\begin{aligned} A_{ij}(t) \\ = \int_{\Omega(t)} \left(\psi_i(\mathbf{x}, t) \nabla \psi_j(\mathbf{x}, t) - \psi_j(\mathbf{x}, t) \nabla \psi_i(\mathbf{x}, t) \right) dx^d \end{aligned} \qquad (78)$$

Now, we recall Eq. (11) obtained during the MLS-SPH-ALE derivation process. Since MLS-SPH-ALE was derived for the Navier–Stokes equations, the ALE flux tensor $\mathbf{F}_w$ and the diffuse tensor $\mathbf{D}$ were grouped in a flux tensor $\mathbf{F}$ defined by $\mathbf{F} \equiv \mathbf{F}_w - \mathbf{D}$. Thus, for the Euler equations $\mathbf{F} = \mathbf{F}_w$, and the MLS-SPH-ALE method provides the following discretization

$$\begin{aligned} \frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} &\left[\frac{1}{2} \left((\mathbf{F}_w)_j + (\mathbf{F}_w)_i \right) - (\mathbf{F}_w)_i \right] \\ &\left(\int_{\partial \Omega} N_i N_j \cdot \mathbf{n} \, dx^{d-1} - \int_{\Omega} N_j \cdot \nabla N_i \, dx^d \right. \\ &\left. + \int_{\Omega} N_i \cdot \nabla N_j \, dx^d \right) = V_i S_i \end{aligned} \qquad (79)$$

Neglecting the boundary term $\left(\int_{\partial \Omega} N_i N_j \cdot \mathbf{n} \, dx^{d-1} \right)$ we note that the MLS-SPH-ALE method uses an interparticle area given by

$$A_{ij} = \int_{\Omega} N_i \cdot \nabla N_j \, dx^d - \int_{\Omega} N_j \cdot \nabla N_i \, dx^d \qquad (80)$$

By comparison of Eq. (78) and Eq. (80) it can be observed that the effective areas resulting from both methods adopt the same structure. The only difference is that A_{ij} for FVPM is given in terms of $\psi_i(\mathbf{x}, t)$ by Eq. (78), meanwhile A_{ij} for MLS-SPH-ALE is given in terms of $N_i(\mathbf{x}, t)$. Since $N_i(\mathbf{x}, t)$ can be obtained with MLS approximation using any set of polynomial basis functions and $\psi_i(\mathbf{x}, t)$ correspond to a particular MLS approximation with constant basis, it can be stated that the expression for A_{ij} obtained with the MLS-SPH-ALE method is more general than the expression obtained with the FVPM method.

FVPM and MLS-SPH-ALE method use different approximations to compute the effective areas. For FVPM, different quadrature rules have been proposed to perform the integration of A_{ij} [57], whereas MLS-SPH-ALE method approximates the effective area Eq. (80) by using a one point quadrature in the overlapping region, resulting in a numerical approximation for A_{ij} as

$$A_{ij} \approx \left(V_i \nabla N_{ji} - V_j \nabla N_{ij} \right) \qquad (81)$$

Table 2 Kernel and MLS approximations for a scalar function and its gradient

Kernel approximation

$\langle u(\boldsymbol{x}) \rangle = \sum_j u(\boldsymbol{x}_j) W(\boldsymbol{x} - \boldsymbol{x}_j, h) V_j$

$\langle \nabla u(\boldsymbol{x}) \rangle = \sum_j u(\boldsymbol{x}_j) \nabla W(\boldsymbol{x} - \boldsymbol{x}_j, h) V_j$

MLS approximation

$\hat{u}(\boldsymbol{x}) = \sum_j N(\boldsymbol{x} - \boldsymbol{x}_j, h) u(\boldsymbol{x}_j)$

$\widehat{\nabla u}(\boldsymbol{x}) = \sum_j \nabla N(\boldsymbol{x} - \boldsymbol{x}_j, h) u(\boldsymbol{x}_j)$

5.3 The SPH-ALE Method

The semi-discrete form of the SPH-ALE method [17] reads as

$$\frac{d}{dt}\big(\boldsymbol{U}_i V_i\big)$$
$$+ V_i \sum_{j=1}^{n_i} V_j \Big(\big(\boldsymbol{F}_{\boldsymbol{w}}^{\alpha}\big)_i + \big(\boldsymbol{F}_{\boldsymbol{w}}^{\alpha}\big)_j \Big) \frac{\partial}{\partial x_i^{\alpha}} W_{ij} = \boldsymbol{S}_i V_i$$

where the fluxes in ALE form were introduced as vectors $\boldsymbol{F}_{\boldsymbol{w}}^{\alpha}$ with the subindex $\boldsymbol{w}$ remarking the ALE character of the flux with respect to a transport velocity field. To ease the comparison with other meshless formulations the ALE flux will be expressed as a tensor, reading as

$$\frac{d}{dt}\big(\boldsymbol{U}_i V_i\big)$$
$$+ V_i \sum_{j=1}^{n_i} V_j \Big(\big(\boldsymbol{F}_{\boldsymbol{w}}\big)_i + \big(\boldsymbol{F}_{\boldsymbol{w}}\big)_j \Big) \nabla_i W_{ij} = \boldsymbol{S}_i V_i \qquad (82)$$

5.3.1 SPH-ALE and MLS-SPH-ALE

Recovering the SPH-ALE method from the MLS-SPH-ALE requires to relate the gradient of the particle shape function ∇N with the gradient of the kernel ∇W.

The semi-discrete-form obtained with MLS-SPH-ALE method for the Euler equations far from the boundary was given in Eq. (14) and it is recalled here for the sake of clarity

$$\frac{d(V_i \boldsymbol{U}_i)}{dt} + \sum_{j=1}^{n_i} \left[\frac{1}{2}\big((\boldsymbol{F}_{\boldsymbol{w}})_j + (\boldsymbol{F}_{\boldsymbol{w}})_i\big) - (\boldsymbol{F}_{\boldsymbol{w}})_i \right]$$
$$\big(-V_j \nabla N_{ij} + V_i \nabla N_{ji}\big) = V_i \boldsymbol{S}_i \qquad (83)$$

In Table 2 the expressions for the kernel and MLS approximations for the function and the gradient of a scalar function u are given

Introducing the approximation $\langle \nabla u(\boldsymbol{x}) \rangle \approx \widehat{\nabla u}(\boldsymbol{x})$, we obtain the following relation

$$\nabla W(\boldsymbol{x} - \boldsymbol{x}_j, h) V_j \approx \nabla N(\boldsymbol{x} - \boldsymbol{x}_j, h)$$

where the kernel and particle shape functions are both centered at $\boldsymbol{x}_j$, and the gradients are taken with respect to the variable $\boldsymbol{x}$. If now the gradients are evaluated at point $\boldsymbol{x}_i$, we obtain the relation

$$\nabla_i W_{ij} V_j \equiv \nabla_i W(\boldsymbol{x}_i - \boldsymbol{x}_j, h) V_j \approx \nabla N(\boldsymbol{x}_i - \boldsymbol{x}_j, h) \equiv \nabla N_{ji} \qquad (84)$$

In Eq. (84) we have introduced the concise notation that is used for the SPH kernel gradient and for particle shape functions. Note that the common nomenclature for the kernel gradient introduces a particle subindex in the nabla operator remarking that the gradient is taken with respect to the spatial coordinates of that particle. Similarly, if the kernel/shape function is centered in particle i and evaluate its gradient in the position of particle j we obtain

$$\nabla_j W_{ji} V_i \equiv \nabla_j W(\boldsymbol{x}_j - \boldsymbol{x}_i, h) V_i \approx \nabla N(\boldsymbol{x}_j - \boldsymbol{x}_i, h) \equiv \nabla N_{ij} \qquad (85)$$

We can insert Eqs. (84) and (85) in Eq. (83) to obtain

$$\frac{d(V_i \boldsymbol{U}_i)}{dt} + V_i \sum_{j=1}^{n_i} V_j \left[\frac{1}{2}\big((\mathbf{F}_{\boldsymbol{w}})_j + (\mathbf{F}_{\boldsymbol{w}})_i\big) - (\mathbf{F}_{\boldsymbol{w}})_i \right]$$
$$\big(-\nabla_j W_{ji} + \nabla_i W_{ij}\big) = V_i \boldsymbol{S}_i$$

Finally, the symmetry property of the kernel $\nabla_j W_{ji} = -\nabla_i W_{ij}$ enable us to write

$$\frac{d(V_i \boldsymbol{U}_i)}{dt} + V_i \sum_{j=1}^{n_i} 2 V_j \left[\frac{1}{2}\big((\mathbf{F}_{\boldsymbol{w}})_j + (\mathbf{F}_{\boldsymbol{w}})_i\big) \right.$$
$$\left. - (\mathbf{F}_{\boldsymbol{w}})_i \right] \nabla_i W_{ij} = V_i \boldsymbol{S}_i \qquad (86)$$

Comparison of Eq. (86) with Eq. (82) shows that the MLS-SPH-ALE method can provide the same semi-discrete form of SPH-ALE method if the interaction flux is given by $\left[\frac{1}{2}\big((\mathbf{F}_{\boldsymbol{w}})_j + (\mathbf{F}_{\boldsymbol{w}})_i\big) \right]$ instead of $\left[\frac{1}{2}\big((\mathbf{F}_{\boldsymbol{w}})_j + (\mathbf{F}_{\boldsymbol{w}})_i\big) - (\mathbf{F}_{\boldsymbol{w}})_i \right]$. Since this self-contribution of particle i could be incorporated or removed in the MLS-SPH-ALE procedure it has been proved that the SPH-ALE method is contained in the MLS-SPH-ALE method.

An interesting analysis is related with the computational cost and accuracy associated with both formulations. MLS-SPH-ALE demands the calculation of gradients ∇N_{ji}, meanwhile SPH-ALE requires the calculation of the terms $\nabla_i W_{ij} V_j$. The calculation of ∇N_{ji} depends on the number of points considered for the support domain of the particle and the dimension of the base of polynomials considered for the MLS approximation. The evaluation of $V_j \nabla_i W_{ij}$ only involves the data associated with particle i and j. However, in terms of accuracy, the SPH-ALE method cannot provide a correct gradient approximation for a linear field. In order to circumvent this flaw, SPH-ALE needs a renormalization correction,

which also demands the solution of a linear system of equations for each particle, as the MLS approximation does.

5.4 The Gizmo-MFV Method

We outline here the Meshless Finite Volume (MFV) method in the form given by Hopkins [1]. The publication is associated with the open source project Gizmo, and therefore we use the name Gizmo-MFV to refer to this method. In [1], the author declares that the derivation of the method is inspired in the work by Gaburov and Nitadori [24]. This derivation procedure was presented for a scalar conservation law meanwhile the procedure presented by Hopkins concerns the Euler equations. Gaburov and Nitadori consider that their derivation is heuristic. More details about this method can be found in [47, 48].

The semi-discrete form obtained for the Gizmo-MFV method for the Euler equations reads as:

$$\frac{d(U_i V_i)}{dt} + \sum_j \left[V_i (F_w^\alpha)_i \tilde{\psi}_j^\alpha(x_i) - V_j (F_w^\alpha)_j \tilde{\psi}_i^\alpha(x_j) \right] = V_i S_i \tag{87}$$

One distinctive feature of the Gizmo-MFV discretization is that it estimates the gradients using a renormalized gradient approximation [21]. The renormalization process guarantees that the discrete approximation reproduces exactly a linear field function. Hopkins [1] and Gaburov and Nitadori [24] use index notation to express the discrete renormalized gradient approximation by

$$(\nabla \varphi)_i^\alpha \approx (D_h^\alpha \varphi)_i = \sum_j (\varphi_j - \varphi_i) \tilde{\psi}_j^\alpha(x_i) \tag{88}$$

where $\tilde{\psi}_j^\alpha(x_i)$ are defined in terms of the renormalization matrix $(B_H^{\alpha,\beta})_i$, particle positions and particle function $\psi_j(x_i)$ to provide a compact expression for the renormalized approximation for the gradient of a field. They can be written as

$$\tilde{\psi}_j^\alpha(x_i) \equiv (B_H^{\alpha,\beta})_i (x_j^\beta - x_i^\beta) \psi_j(x_i) \tag{89}$$

The renormalization matrix $(B_H^{\alpha,\beta})_i$ is required to satisfy condition

$$\delta^{\alpha,\gamma} = \sum_j (B_H^{\alpha,\beta})_i (x_j^\beta - x_i^\beta)(x_j^\gamma - x_i^\gamma) \psi_j(x_i) \tag{90}$$

where $\delta^{\alpha,\gamma}$ is the Kronecker delta function. Since the definition of the renormalization matrix $(B_H^{\alpha,\beta})_i$ given by Eq. (90) is taken from Hopkins [1] we append the subindex H in the designation of the matrix.

The Gizmo-MFV discretization replaces the particle fluxes $(F_w)_i$ and $(F_w)_j$ with the numerical flux solution of the moving

Riemann problem defined by the states of particles/cells i and j. Defining such a flux as $(G_w)_{ij}$, Eq. (87) is reinterpreted as

$$\frac{d(U_i V_i)}{dt} + \sum_j (G_w)_{ij} \cdot \left[V_i \tilde{\psi}_j(x_i) - V_j \tilde{\psi}_i(x_j) \right] = V_i S_i \tag{91}$$

where we have changed from the index notation for the spatial components to the more compact vector notation.

Transition from Eqs. (87) to (91) is not supported by mathematical operations. The formal pass from Eq. (87) to Eq. (91) demands that $(F_w^\alpha)_i = (F_w^\alpha)_j = (F_w^\alpha)_{ij}$ and this condition is only verified for uniform flows. Hopkins and Gaburov motivate the introduction of the solution of the Riemann problem to automatically include the dissipation terms to stabilize a central scheme.

Comparing Eqs. (91) with (46) it can be inferred the definition of an effective interaction area $A_{ij}(t)$, between particles i and j, which is given by

$$A_{ij} \equiv V_i \tilde{\psi}_j(x_i) - V_j \tilde{\psi}_i(x_j) \tag{92}$$

5.4.1 Gizmo-MFV and MLS-SPH-ALE

Now we compare the Gizmo-MFV and MLS-SPH-ALE discretization of Euler equations. For the sake of clarity, we recall that Gizmo-MFV method is given by Eq. (91)

$$\frac{d(U_i V_i)}{dt} + \sum_j (G_w)_{ij} \cdot \left[V_i \tilde{\psi}_j(x_i) - V_j \tilde{\psi}_i(x_j) \right] = V_i S_i$$

whereas the discretization of the MLS-SPH-ALE method particularized for the Euler equations is given by Eq. (14) as

$$\frac{d(V_i U_i)}{dt} + \sum_{j=1}^{n_i} \left[\frac{1}{2} \left((F_w)_j + (F_w)_i \right) - (F_w)_i \right]$$
$$(-V_j \nabla N_{ij} + V_i \nabla N_{ji}) = V_i S_i$$

By identification of the effective fluxes and interaction areas we can obtain the particular settings of the MLS-SPH-ALE formulation to recover the Gizmo-MFV discretization.

The effective flux of Gizmo-MFV is given by $(G_w)_{ij}$, which is obtained as the solution of a Riemann problem at a point between particles. In MLS-SPH-ALE, the effective flux reads as $\left[\frac{1}{2} \left((F_w)_j + (F_w)_i \right) - (F_w)_i \right]$. For stability reasons the central flux $\frac{1}{2} \left((F_w)_j + (F_w)_i \right)$ is replaced by the solution of the Riemann problem $\frac{1}{2} \left((F_w)_j + (F_w)_i \right) \approx (G_w)_{ij}$. Therefore, the only difference in the flux expression resides in the subtraction of flux of particle $(F_w)_i$ and this term can optionally be incorporated or removed in the MLS-SPH-ALE procedure.

Since the effective interaction areas for the Gizmo-MFV method are defined by Eq. (92), the condition for equivalent

effective interaction areas demands a particular setting in the MLS approximation giving

$$\nabla N(\boldsymbol{x}_i - \boldsymbol{x}_j, h) = \nabla N_{ji} = \tilde{\psi}_j^{\alpha}(\boldsymbol{x}_i) \tag{93}$$

We now look for the corresponding approximation for the gradient of a field by using MLS approximations. We start by expressing the MLS approximation of scalar field φ as

$$\hat{\varphi}(\boldsymbol{x}) = \sum_j N(\boldsymbol{x} - \boldsymbol{x}_j, h)\varphi(\boldsymbol{x}_j) \tag{94}$$

then we take the gradient to previous expression to provide the MLS gradient approximation as

$$\nabla(\hat{\varphi}(\boldsymbol{x})) = \nabla\hat{\varphi}(\boldsymbol{x}) = \sum_j \nabla N(\boldsymbol{x} - \boldsymbol{x}_j, h)\varphi(\boldsymbol{x}_j) \tag{95}$$

Using the partition of nullity property for a particular point $\boldsymbol{x}$ where the gradient is to be computed, the previous approximation for the gradient can also be rewritten as

$$\nabla\hat{\varphi}(\boldsymbol{x}) = \sum_j \nabla N(\boldsymbol{x} - \boldsymbol{x}_j, h)(\varphi(\boldsymbol{x}_j) - \varphi(\boldsymbol{x})) \tag{96}$$

that evaluated in the particular point $\boldsymbol{x} = \boldsymbol{x}_i$ yields

$$\nabla\hat{\varphi}(\boldsymbol{x}_i) = \sum_j \nabla N(\boldsymbol{x}_i - \boldsymbol{x}_j, h)(\varphi(\boldsymbol{x}_j) - \varphi(\boldsymbol{x}_i)) \tag{97}$$

By comparing Eq. (88) and Eq. (97), it can be inferred that ∇N_{ji} and $\tilde{\psi}_j(\boldsymbol{x}_i)$ play the same role as approximations of the gradient of a scalar field.

Using Eq. (97) in Eq. (88), we can write

$$\begin{aligned}(\nabla\varphi)_i &\approx \nabla\hat{\varphi}(\boldsymbol{x}_i) \\ &= \sum_j \nabla N(\boldsymbol{x}_i - \boldsymbol{x}_j, h)(\varphi(\boldsymbol{x}_j) - \varphi(\boldsymbol{x}_i)) \\ &= \sum_j \nabla N_{ji}(\varphi_j - \varphi_i)\end{aligned} \tag{98}$$

However, we have not proved yet that $\tilde{\psi}_j(\boldsymbol{x}_i) = \nabla N_{ji}$. In the particular case of linear fields, the renormalized gradient derivative and the MLS approximations (with at least a linear basis) reproduce exactly a linear field and therefore $(D_h\varphi)_i = \nabla\hat{\varphi}(\boldsymbol{x}_i)$ and we can conclude that ∇N_{ji} and $\tilde{\psi}_j^{\alpha}(\boldsymbol{x}_i)$ for this particular field.

In order to ease the comparison with other formulations it is convenient to provide an alternative expression for $\tilde{\psi}_i(\boldsymbol{x}_j)$. Gizmo-MFV defines $\tilde{\psi}_i(\boldsymbol{x}_j)$ in Eq. (88) in terms of a renormalization matrix $B_H^{\alpha,\beta}$ satisfying condition given by Eq. (90). The matrix $B_H^{\alpha,\beta}$ is not the common renormalization matrix used in the SPH community. For instance, Lanson and Vila [21] define other renormalization matrix $(B_V^{\alpha,\beta})_i$ forcing the fulfillment of condition

$$\delta^{\alpha,\gamma} = \sum_{j \in} \left(B_V^{\alpha,\beta}\right)_i \frac{\partial W(\boldsymbol{x}_i - \boldsymbol{x}_j, h)}{\partial x_i^{\beta}}\left(x_j^{\gamma} - x_i^{\gamma}\right)V_j \tag{99}$$

The matrix used by Gizmo-MFV $B_H^{\alpha,\beta}$ is different from the matrix used in renormalized SPH-ALE $B_V^{\alpha,\beta}$. Even more, $B_V^{\alpha,\beta}$ is dimensionless but $B_H^{\alpha,\beta}$ has dimensions of $[L]^{-2}$. The renormalization matrix $B_H^{\alpha,\beta}$ has the computational advantage of operating over symmetric matrices.

Since the Kronecker function $\delta^{\alpha,\gamma}$ appears on both left hand side of Eq. (90) and Eq. (99), the following relation between matrices is obtained

$$\underbrace{\left(B_H^{\alpha,\beta}\right)_i\left(x_j^{\beta} - x_i^{\beta}\right)\psi_j(\boldsymbol{x}_i)}_{\tilde{\psi}_j^{\alpha}(\boldsymbol{x}_i)} = \left(B_V^{\alpha,\beta}\right)_i \frac{\partial W(\boldsymbol{x}_i - \boldsymbol{x}_j, h)}{\partial x_i^{\beta}}V_j \tag{100}$$

where the left hand side of Eq. (100) corresponds with the definition of $\tilde{\psi}_j^{\alpha}(\boldsymbol{x}_i)$ in Eq. (89). Therefore we have obtained and alternative definition of $\tilde{\psi}_j^{\alpha}(\boldsymbol{x}_i)$ in terms of the kernel gradient $\nabla_i W_{ij}$

$$\tilde{\psi}_j^{\alpha}(\boldsymbol{x}_i) = \left(B_V^{\alpha,\beta}\right)_i \frac{\partial W(\boldsymbol{x}_i - \boldsymbol{x}_j, h)}{\partial x_i^{\beta}}V_j \tag{101}$$

or equivalently,

$$\tilde{\psi}_j(\boldsymbol{x}_i) = \left(B_V\right)_i \nabla_i W_{ij} V_j \tag{102}$$

Introducing Eq. (102) into the semi-discrete form obtained by the Gizmo-MFV given by Eq. (91) gives

$$\begin{aligned}\frac{d(U_i V_i)}{dt} &+ \sum_j (\mathbf{G}_w)_{ij} \cdot \left[V_i(B_V)_i \nabla_i W_{ij} V_j - \right. \\ &\left. V_j(B_V)_j \nabla_j W_{ji} V_i\right] = V_i S_i\end{aligned} \tag{103}$$

Omitting the renormalization process is equivalent to consider that the renormalization matrix is the identity matrix, that is to say, $(B_V)_i = (B_V)_j = \mathbb{I}$. For that particular case the Gizmo-MFV formulation becomes the SPH-ALE formulation of Vila.

5.5 The Finite Point Method (FPM)

In this section we address the analysis of a member of the Group III in the Hopkins' classification. Note that in Sect. 4.5 we have already shown that SPH is closely linked to MFVM and therefore to MLS-SPH-ALE. Here we show that the MLS-SPH-ALE is also a general formulation for those methods of Group III that do not use the kernel approximation. The Finite Point Method (FPM) is a meshless technique that was originally presented by Oñate et al.

[34, 58] and [59]. The solution of compressible Euler equations with FPM have been considered in detail in [60, 61]. Bajko [62] presented an accuracy improvement of the FPM that was applied for computational aeroacoustics.

FPM is based on the strong solution of PDE over a stationary cloud of points. It is a method that does not assign/calculate any weight to the points of the cloud (neither volume nor mass). This kind of meshless methods resembles the FDM in the collocation approach and are usually designated as Generalized Finite Difference Methods (GFDM), because the grid layout can be considered a special case of a point cloud. Both the strong form and the stationary position of the points are important differences with other meshless formulations considered in the present section. The FPM approximates flow variables and their derivatives by means of a Weighted Least-Square procedure that is known as Fixed Least Squares (FLS). The MLS-SPH-ALE method uses an approximation based on Moving Least Squares, but the ALE capability allow us to set the frame velocity $w = 0$, and with fixed point position MLS and FLS provide the same approximations. In the literature there are other GFDMs that share many features with the FPM, but differ in the technique used to approximate the partial derivatives on scattered data.

In the same manner that FDM formulation can be derived from the perspective of FVM we propose to obtain the FPM formulation starting from the MLS-SPH-ALE method. The acronym FPM may be a bit confusing because it is also used for other meshless methods like the Finite Pointset Method [35], and the Finite Particle Method [63] but we recall that in this work it is used for the Finite Point Method. In these two formulations particles can either be fixed in an Eulerian frame as interpolation points or can move in a Lagrangian frame.

We outline the derivation of the FPM following the works of [60] and [61]. The derivation begins from the Euler equations, written in differential conservative form:

$$\frac{\partial U(x,t)}{\partial t} + \frac{\partial F_E^\alpha(x,t)}{\partial x^\alpha} = S(x,t) \tag{104}$$

In order to obtain the semi-discrete form, the collocation method is adopted, so the following expression should be satisfied in each point of the cloud

$$\left[\frac{\partial \hat{U}(x,t)}{\partial t} + \frac{\partial \hat{F}_E^\alpha(x,t)}{\partial x^\alpha}\right]_{x=x_i} = \left[\hat{S}(x,t)\right]_{x=x_i} \tag{105}$$

where hats are used to denote the FLS approximations. We note that FLS shape functions do not interpolate point data, but assuming the approximation for the gradient and source term Eq. (105) becomes

$$\frac{\partial U(x_i,t)}{\partial t} + \left[\frac{\partial \hat{F}_E^\alpha(x,t)}{\partial x^\alpha}\right]_{x=x_i} = S(x_i,t) \tag{106}$$

In order to proceed further, Eq. (106) demands to evaluate the divergence of the flux. Noting that FLS corresponds with a particular case of MLS with fixed point position, the FLS approximation for the Eulerian flux is expressed in terms of $N(x - x_j)$ by

$$\hat{F}_E^\alpha(x,t) = \sum_j N(x - x_j) F_E^\alpha(x_j,t) \tag{107}$$

By taking the divergence of the previous flux expression we obtain

$$\begin{aligned}\frac{\partial \hat{F}_E^\alpha(x,t)}{\partial x^\alpha} &= \sum_j \frac{\partial N(x - x_j)}{\partial x^\alpha} F_E^\alpha(x_j,t) \\ &= \sum_j \nabla N(x - x_j) \mathbf{F}_E(x_j,t)\end{aligned} \tag{108}$$

At this point, the Partition of Nullity property $\sum_j \nabla N(x - x_j) = 0$ is invoked and a null term in the flux divergence is subtracted

$$\begin{aligned}\frac{\partial \hat{F}_E^\alpha(x,t)}{\partial x^\alpha} &= \sum_j \nabla N(x - x_j)\big(\mathbf{F}_E(x_j,t) - \mathbf{F}_E(x,t)\big)\end{aligned} \tag{109}$$

We can now evaluate in the position of point i giving

$$\begin{aligned}\left[\frac{\partial \hat{F}_E^\alpha(x,t)}{\partial x^\alpha}\right]_{x=x_i} &= \sum_j \nabla N(x_i - x_j)\big(\mathbf{F}_E(x_j,t) - \mathbf{F}_E(x_i,t)\big)\end{aligned} \tag{110}$$

By using the compact notation $\nabla N(x_i - x_j) \equiv \nabla N_j(x_i) \equiv \nabla N_{ji}$ and inserting Eq. (110) into Eq. (106) the following semi-discrete expression is obtained

$$\begin{aligned}\frac{\partial U(x_i,t)}{\partial t} &+ \sum_j \nabla N_{ji}\bigg(\mathbf{F}_E(x_j,t) - \mathbf{F}_E(x_i,t)\bigg) = S(x_i,t)\end{aligned} \tag{111}$$

This semi-discrete form is not stable. In order to stabilize the scheme, the following approximation is used

$$\mathbf{F}_E(x_j,t) - \mathbf{F}_E(x_i,t) \approx \\ 2\bigg(\mathbf{F}_E\bigg(\frac{x_i + x_j}{2},t\bigg) - \mathbf{F}_E(x_i,t)\bigg)$$

Or, in a more compact notation

$$(\mathbf{F}_E)_j - (\mathbf{F}_E)_i \approx 2\big((\mathbf{F}_E)_{ij} - (\mathbf{F}_E)_i\big) \tag{112}$$

Since $(\mathbf{F}_E)_{ij}$ is the flux in the midpoint of the segment connecting points i and j, it could be numerically evaluated with an approximate Riemann solver introducing the required dissipation, that is, $(\mathbf{F}_E)_{ij} \approx (\mathbf{G}_E)_{ij}$. Inserting the approximation given by Eq. (112) into Eq. (111) the final semi-discrete equation for the FPM read as

$$\frac{\partial \boldsymbol{U}_i}{\partial t} + 2 \sum_j \nabla N_{ji}\big((\mathbf{G}_E)_{ij} - (\mathbf{F}_E)_i\big) = \boldsymbol{S}_i \tag{113}$$

where it can be observed that neither mass nor volume associated with the point appear in the semi-discrete form.

5.5.1 FPM and MLS-SPH-ALE

Starting with the MLS-SPH-ALE, the semi-discrete form for the Euler equations is given by Eq. (14), that we recall here for convenience

$$\frac{d(V_i \boldsymbol{U}_i)}{dt} + \sum_{j=1}^{n_i} \left[\frac{1}{2}\big((\boldsymbol{F}_w)_j + (\boldsymbol{F}_w)_i\big) - (\boldsymbol{F}_w)_i \right]$$
$$\big(-V_j \nabla N_{ij} + V_i \nabla N_{ji}\big) = V_i \boldsymbol{S}_i$$

For MLS-SPH-ALE method running in Eulerian mode the transport velocity $\boldsymbol{w} = \boldsymbol{0}$ and therefore $\mathbf{F}_w = \mathbf{F}_E - \boldsymbol{w} \otimes \boldsymbol{U} = \mathbf{F}_E$. In Eulerian framework the time derivative $\dfrac{d(V_i \boldsymbol{U}_i)}{dt}$ is identical to $\dfrac{\partial(V_i \boldsymbol{U}_i)}{\partial t}$ and since particles volumes do not vary in time $\dfrac{\partial(V_i \boldsymbol{U}_i)}{\partial t} = V_i \dfrac{\partial \boldsymbol{U}_i}{\partial t}$. Taking into account these considerations, the final semi-discrete system obtained with MLS-SPH-ALE method for the Euler equations in Eulerian framework reads as

$$V_i \frac{\partial(\boldsymbol{U}_i)}{\partial t} + \sum_{j=1}^{n_i} \left[\frac{1}{2}\big((\mathbf{F}_E)_j + (\mathbf{F}_E)_i\big) - (\mathbf{F}_E)_i \right]$$
$$\big(-V_j \nabla N_{ij} + V_i \nabla N_{ji}\big) = V_i \boldsymbol{S}_i \tag{114}$$

By comparison of Eq. (113) and Eq. (114) it is inferred that we need to introduce particular settings in the MLS-SPH-ALE method that allow us to cancel the volume contributions. In order to factorize the volume V_i in the expression for the flux we can invoke several simplifications.

The simplest case considers that nodes are disposed in a uniform Cartesian grid. In that case $V_i = V_j$ and by symmetry $\nabla N_{ij} = -\nabla N_{ji}$. Inserting these relations in Eq. (114) the semi-discrete form becomes

$$\frac{\partial \boldsymbol{U}_i}{\partial t}$$
$$+ \sum_{j=1}^{n_i} \left[\frac{1}{2}\big((\mathbf{F}_E)_j + (\mathbf{F}_E)_i\big) - (\mathbf{F}_E)_i \right]\big(2 \nabla N_{ji}\big) = \boldsymbol{S}_i \tag{115}$$

that provides the same expression than that obtained for the FPM in Eq. (113) once the central flux $\frac{1}{2}\big((\mathbf{F}_E)_j + (\mathbf{F}_E)_i\big)$ is approximated by a numerical flux $(\mathbf{G}_E)_{ij}$. However, this particular case is not very interesting since a meshless formulation with points placed in a regular grid does not provide any advantage over mesh-based methods.

We consider now a more general case with a fixed cloud of points with irregular distribution. In that scenario the volumes do not evolve in time and the volumes associated to particles may vary from one particle to another. Returning to Eq. (114) we focus the attention on the term inside brackets $\big(-V_j \nabla N_{ij} + V_i \nabla N_{ji}\big)$. The gradients of the shape functions N_{ij} and N_{ji} can be expressed in terms of the kernel gradient, according to the approximations given in Eq. (84) and Eq. (85)

$$\nabla_i W_{ij} V_j \approx \nabla N_{ji} \qquad \nabla_j W_{ji} V_i \approx \nabla N_{ij} \tag{116}$$

Inserting the approximations given by Eq. (116) into Eq. (114) enables us to cancel the term V_i leading to

$$\frac{\partial(\boldsymbol{U}_i)}{\partial t} + \sum_{j=1}^{n_i} \left[\frac{1}{2}\big((\mathbf{F}_E)_j + (\mathbf{F}_E)_i\big) - (\mathbf{F}_E)_i \right]$$
$$\big(-V_j \nabla_j W_{ji} + \nabla_i W_{ij} V_j\big) = \boldsymbol{S}_i \tag{117}$$

since the kernel gradient is symmetric we can express the expression inside brackets of Eq. (117) as $2\nabla_i W_{ij} V_j$. Using the approximation $\nabla N_{ji} \approx \nabla_i W_{ij} V_j$ given in the left side of Eq. (116) we arrive at the same semi-discrete system given by Eq. (115), but for a general cloud of points that does not need to fulfill the conditions $V_i = V_j$ and symmetry $\nabla N_{ij} = -\nabla N_{ji}$.

6 Conclusions

Meshless methods have not yet attained the same level of development as mesh-based methods. However, these methods are receiving great attention in different research communities. These conditions gave rise to a myriad of meshless methods derived by different groups. Comparison between different formulations reveals that although procedures could vary in great extent, the final semi-discrete equations have great similarities.

In this work we have revisited the derivation of a set of meshless formulations and we have shown the examined methods can be obtained from particular settings of the MLS-SPH-ALE discretization. We note that boundary conditions treatment has been excluded in this work. However, in the derivation procedure we can clearly see that some methods ease the implementation of the boundary conditions with a boundary flux term. Thus, in the MLS-SPH-ALE method the boundary condition are imposed thought the fluxes. We have also shown that MLS-SPH-ALE is a general formulation that includes many of the examined methods. In our opinion, the knowledge about the connections among different numerical schemes, apparently without any relation between them, can help the research community to find new more accurate and more efficient numerical schemes.

We have shown the relations between the three groups of the Hopkins classification, and we have seen how the MFVM relates with FVM and SPH. Moreover, we have proved that the MFVM provides a semi-discrete form where the interaction flux is not fully discretized in the particle positions. By assuming a constant interaction flux in the MFVM we obtain the same continuous effective area that the one obtained in MLS-SPH-ALE method.

The MLS-SPH-ALE method has the advantages of a very flexible selection of particle shape functions. The FVPM was derived by using the derivation properties of the simplest shape function satisfying the partition of unity property. The shape functions used by the FVPM correspond with the MLS shape functions using a polynomial basis with a constant term. Moreover, The FVPM is the procedure that gives the clearest interpretation of the interaction flux in the overlap region of particles. In the MLS-SPH-ALE method the procedure to define this effective flux is not unique.

SPH-ALE and Gizmo-MFV methods obtain a discrete effective area in terms of the gradient of the kernel and the volume associated to particles. In the MLS-SPH-ALE method, the effective area is a symmetric expression given in terms of the gradient of the particle shape functions and the volume of the particles. MLS shape functions are calculated by means of a weighted least squares approach. When the weighted function used by MLS is exactly the kernel function we can obtain an analytic expression relating the kernel gradient with the shape function gradient. However, in general we cannot provide an analytic expression to relate the kernel gradient with the gradient of a particle shape function obtained with MLS. By comparison of both, the kernel and MLS approximations of the gradient of a function, we manage to make some term identifications to prove that MLS-SPH-ALE can be particularized to arrive at the semi-discrete equations derived with SPH-ALE and Gizmo-MFV methods.

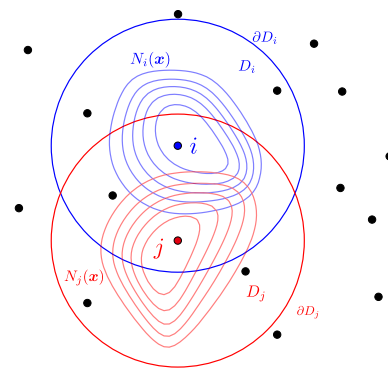Finally, we have traced a path relating the MLS-SPH-ALE method with the Finite Point Method.

Fig. 2 Isoline representation of particle shape functions N_i and N_j for two interacting particles i and j. Positions of particles i and j are represented by blue and red filled circles respectively. Other particles positions are represented by black filled circles

Appendix 1: Partition of Unity

A set of shape functions verifies the Partition of Unity (PU) when the following property holds for any point x in the domain Ω

$$\sum_j N(x - x_j, h) = \sum_{j=1}^{n_x} N(x - x_j, h) = 1 \tag{A1}$$

where n_x denotes the number of neighbors in a compact support centered in x and extension related with length h. PU property provides a subdivision of the domain Ω into a set of subdomains Ω_i. Each subdomain Ω_i is associated with a particle i and a shape function $N(x - x_i, h)$ that is non-zero only in that subdomain.

Figure 2 provides a schematic representation of two particle shape functions $N_i = N(x - x_i, h)$ and $N_j = N(x - x_j, h)$. Some isoline values are represented to remark that shape functions do not have the symmetry property. Since the shape functions are usually associated with a kernel o weighted function we also depict the support domain D and its border ∂D.

The PU property assures the zeroth-order consistency meaning that any constant function c can be reproduced exactly

$$\hat{u}(x) = \sum_j N_j(x) u_j = c \sum_{j=1}^{n_x} N(x - x_j) = c \tag{A2}$$

By taking the gradient operator at both sides of $\sum_{j=1}^{n_x} N(x - x_j) = 1$ and noting that ∇ is a linear operator it can be concluded that the derivatives of the shape functions verify also the Partition of Nullity

$$\nabla \sum_{j=1}^{n_x} N(\boldsymbol{x} - \boldsymbol{x}_j) = \nabla 1$$

$$\sum_{j=1}^{n_x} \nabla N(\boldsymbol{x} - \boldsymbol{x}_j) = 0 \tag{A3}$$

It is possible to take advantage of the PU property when it is necessary to conduct integrals over the global domain Ω or over any subdomain Ω_i.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Hopkins PF (2015) A new class of accurate, mesh-free hydrodynamic simulation methods. Mon Not R Astron Soc 450(1):53–110. https://doi.org/10.1093/mnras/stv195
2. Chen JS, Hillman M, Chi SW (2017) Meshfree methods: progress made after 20 years. J Eng Mech 143(4):04017001. https://doi.org/10.1061/(asce)em.1943-7889.0001176
3. Huerta A, Belytschko T, Fernández-Méndez S, Rabczuk T, Zhuang X, Arroyo M (2017) Meshfree methods. In: Encyclopedia of computational mechanics, 2nd edn. Wiley, New York, p 38
4. Monaghan JJ (1989) On the problem of penetration in particle methods. J Comput Phys 82(1):1–15. https://doi.org/10.1016/0021-9991(89)90032-6
5. Adami S, Hu XY, Adams NA (2013) A transport-velocity formulation for smoothed particle hydrodynamics. J Comput Phys 241:292–307. https://doi.org/10.1016/j.jcp.2013.01.043
6. Oger G, Marrone S, Touzé DL, de Leffe M (2016) SPH accuracy improvement through the combination of a quasi-Lagrangian shifting transport velocity and consistent ALE formalisms. J Comput Phys 313:76–98. https://doi.org/10.1016/j.jcp.2016.02.039
7. Antuono M, Sun PN, Marrone S, Colagrossi A (2021) The δ-ALE-SPH model: an arbitrary lagrangian-eulerian framework for the δ-SPH model with particle shifting technique. Comput Fluids 216:104806. https://doi.org/10.1016/j.compfluid.2020.104806
8. Cercos-Pita JL, Dalrymple RA, Herault A (2016) Diffusive terms for the conservation of mass equation in SPH. Appl Math Model 40(19–20):8722–8736. https://doi.org/10.1016/j.apm.2016.05.016
9. Cueto-Felgueroso L, Colominas I, Nogueira X, Navarrina F, Casteleiro M (2007) Finite volume solvers and Moving Least-Squares approximations for the compressible Navier-Stokes equations on unstructured grids. Comput Methods Appl Mech Eng 196(45–48):4712–4736. https://doi.org/10.1016/j.cma.2007.06.003
10. Ramírez L, Nogueira X, Khelladi S, Chassaing JC, Colominas I (2014) A new higher-order finite volume method based on Moving Least Squares for the resolution of the incompressible Navier-Stokes equations on unstructured grids. Comput Methods Appl Mech Eng 278:883–901. https://doi.org/10.1016/j.cma.2014.06.028
11. Hirsch C (2007) Numerical computation of internal and external flows: the fundamentals of computational fluid dynamics, 2nd edn. Elsevier/Butterworth-Heinemann, Oxford
12. Blazek J (2015) Computational fluid dynamics: principles and applications, 3rd edn. Elseiver/Butterworth-Heinemann, Oxford
13. Zienkiewicz OC, Taylor RL, Nithiarasu P (2014) The finite element method for fluid dynamics, 7th edn. Elsevier/Butterworth-Heinemann, Oxford
14. Springel V (2010) E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh. Mon Not R Astron Soc 401(2):791–851. https://doi.org/10.1111/j.1365-2966.2009.15715.x
15. Muñoz DJ, Springel V, Marcus R, Vogelsberger M, Hernquist L (2012) Multidimensional, compressible viscous flow on a moving Voronoi mesh. Mon Not R Astron Soc 428(1):254–279. https://doi.org/10.1093/mnras/sts015
16. Ivanova N, Justham S, Chen X, Marco OD, Fryer CL, Gaburov E et al (2013) Common envelope evolution: where we stand and how we can move forward. Astron Astrophys Rev 21(1). https://doi.org/10.1007/s00159-013-0059-2
17. Vila JP (1999) On particle weighted methods and smooth particle hydrodynamics. Math Models Methods Appl Sci 09(02):161–209. https://doi.org/10.1142/s0218202599000117
18. Ben Moussa B, Vila JP (2000) Convergence of SPH method for scalar nonlinear conservation laws. SIAM J Numer Anal 37(3):863–887. https://doi.org/10.1137/S0036142996307119
19. Avesani D, Dumbser M, Bellin A (2014) A new class of moving-least-squares WENO-SPH schemes. J Comput Phys 270:278–299. https://doi.org/10.1016/j.jcp.2014.03.041
20. Nogueira X, Ramírez L, Clain S, Loubère R, Cueto-Felgueroso L, Colominas I (2016) High-accurate SPH method with multidimensional optimal order detection limiting. Comput Methods Appl Mech Eng 310:134–155. https://doi.org/10.1016/j.cma.2016.06.032
21. Lanson N, Vila JP (2008) Renormalized meshfree schemes I: consistency, stability, and hybrid methods for conservation laws. SIAM J Numer Anal 46(4):1912–1934. https://doi.org/10.1137/S0036142903427718

22. Lanson N, Vila JP (2008) Renormalized meshfree schemes II: convergence for Scalar Conservation Laws. SIAM J Numer Anal 46(4):1935–1964. https://doi.org/10.1137/S003614290444739X

23. Hietel D, Steiner K, Struckmeier J (2000) A finite-volume particle method for compressible flows. Math Models Methods Appl Sci 10(09):1363–1382. https://doi.org/10.1142/s0218202500000604

24. Gaburov E, Nitadori K (2011) Astrophysical weighted particle magnetohydrodynamics. Mon Not R Astron Soc 414(1):129–154. https://doi.org/10.1111/j.1365-2966.2011.18313.x

25. Ramírez L, Nogueira X, Khelladi S, Krimi A, Colominas I (2018) A very accurate arbitrary Lagrangian-Eulerian meshless method for computational aeroacoustics. Comput Methods Appl Mech Eng 342:116–141. https://doi.org/10.1016/j.cma.2018.07.036

26. Ramírez L, Eirís A, Couceiro I, París J, Nogueira X (2022) An arbitrary Lagrangian-Eulerian SPH-MLS method for the computation of compressible viscous flows. J Comput Phys. https://doi.org/10.1016/j.jcp.2022.111172

27. Dilts GA (1999) Moving-least-squares-particle hydrodynamics I. Consistency and stability. Int J Numer Methods Eng 44(8):1115–1155. https://doi.org/10.1002/(sici)1097-0207(19990320)44:8<1115::aid-nme547>3.0.co;2-l

28. Dilts GA (2000) Moving least-squares particle hydrodynamics II: conservation and boundaries. Int J Numer Methods Eng 48(10):1503–1524. https://doi.org/10.1002/1097-0207(20000810)48:10<1503::aid-nme832>3.0.co;2-d

29. Monaghan JJ (1988) An introduction to SPH. Comput Phys Commun 48(1):89–96. https://doi.org/10.1016/0010-4655(88)90026-4

30. Monaghan JJ (2005) Smoothed particle hydrodynamics. Rep Prog Phys 68(8):1703–1759. https://doi.org/10.1088/0034-4885/68/8/R01

31. Colagrossi A, Landrini M (2003) Numerical simulation of interfacial flows by smoothed particle hydrodynamics. J Comput Phys 191(2):448–475. https://doi.org/10.1016/S0021-9991(03)00324-3

32. Inutsuka S (2002) Reformulation of smoothed particle hydrodynamics with Riemann solver. J Comput Phys 179(1):238–267. https://doi.org/10.1006/jcph.2002.7053

33. Parshikov AN, Medin SA (2002) Smoothed particle hydrodynamics using interparticle contact algorithms. J Comput Phys 180(1):358–382. https://doi.org/10.1006/jcph.2002.7099

34. Oñate E, Idelsohn S, Zienkiewicz OC, Taylor RL (1996) A finite point method in computational mechanics. Applications to convective transport and fluid flow. Int J Numer Methods Eng 39(22):3839–3866. https://doi.org/10.1002/(sici)1097-0207(19961130)39:22<3839::aid-nme27>3.0.co;2-r

35. Kuhnert J (2003) An upwind finite pointset method (FPM) for compressible Euler and Navier-Stokes equations. Lecture notes in computational science and engineering. Springer, New York, pp 239–249

36. Suchde P, Kuhnert J, Schröder S, Klar A (2017) A flux conserving meshfree method for conservation laws. Int J Numer Methods Eng 112(3):238–256. https://doi.org/10.1002/nme.5511

37. Few CG, Dobbs C, Pettitt A, Konstandin L (2016) Testing hydrodynamics schemes in galaxy disc simulations. Mon Not R Astron Soc 460(4):4382–4396. https://doi.org/10.1093/mnras/stw1226

38. Schaller M, Bower RG, Theuns T (2013) On the use of particle based methods for cosmological hydrodynamical simulations. In: 8th International SPHERIC workshop, Trondheim, Norway, p 7

39. Nayroles B, Touzot G, Villon P (1992) Generalizing the finite element method: Diffuse approximation and diffuse elements. Comput Mech 10(5):307–318. https://doi.org/10.1007/bf00364252

40. Gaburro E, Boscheri W, Chiocchetti S, Klingenberg C, Springel V, Dumbser M (2020) High order direct Arbitrary-Lagrangian-Eulerian schemes on moving Voronoi meshes with topology changes. J Comput Phys 407:109167. https://doi.org/10.1016/j.jcp.2019.109167

41. Huang TH, Wei H, Chen JS, Hillman MC (2019) RKPM2D: an open-source implementation of nodally integrated reproducing kernel particle method for solving partial differential equations.

42. Price DJ (2012) Smoothed particle hydrodynamics and magnetohydrodynamics. J Comput Phys 231(3):759–794. https://doi.org/10.1016/j.jcp.2010.12.011

43. Rosswog S (2009) Astrophysical smooth particle hydrodynamics. New Astron Rev 53(4–6):78–104. https://doi.org/10.1016/j.newar.2009.08.007

44. Violeau D (2012) Fluid mechanics and the SPH method. Oxford University Press, Oxford

45. Khayyer A, Gotoh H (2010) A higher order Laplacian model for enhancement and stabilization of pressure calculation by the MPS method. Appl Ocean Res 32(1):124–131. https://doi.org/10.1016/j.apor.2010.01.001

46. Junk M (2003) Do finite volume methods need a mesh? Meshfree methods for partial differential equations. Springer, Berlin, pp 223–238

47. Hinz PM (2020) Implementing a meshless finite mass scheme in the cosmological N-body code Gadget. Master Thesis, Ludwig-Maximilians-Universität München, Germany

48. Alonso-Asensio I (2019) Mesh-free hydrodynamics: theory and validation of the method. Master Thesis, Universidad de La Laguna, Spain

49. Junk M, Struckmeier J (2000) Consistency analysis of mesh-free methods for conservation laws. Mitteilungen der Gesellschaft für Angewandte Mathematik und Mechanik. 24(2):99–126

50. Keck R, Hietel D (2005) A projection technique for incompressible flow in the meshless finite volume particle method. Adv Comput Math 23(1–2):143–169. https://doi.org/10.1007/s10444-004-1831-7

51. Teleaga D, Struckmeier J (2008) A finite-volume particle method for conservation laws on moving domains. Int J Numer Methods Fluids 58(9):945–967. https://doi.org/10.1002/fld.1778

52. Nestor RM, Basa M, Lastiwka M, Quinlan NJ (2009) Extension of the finite volume particle method to viscous flow. J Comput Phys 228(5):1733–1749. https://doi.org/10.1016/j.jcp.2008.11.003

53. Quinlan NJ (2018) Extensions of the meshless finite volume particle method (FVPM) for static and dynamic free-surface flows. Comput Fluids 177:33–45. https://doi.org/10.1016/j.compfluid.2018.09.019

54. Jahanbakhsh E, Maertens A, Quinlan NJ, Vessaz C, Avellan F (2017) Exact finite volume particle method with spherical-support kernels. Comput Methods Appl Mech Eng 317:102–127. https://doi.org/10.1016/j.cma.2016.12.015

55. Jahanbakhsh E, Vessaz C, Maertens A, Avellan F (2016) Development of a finite volume particle method for 3-D fluid flow simulations. Comput Methods Appl Mech Eng 298:80–107. https://doi.org/10.1016/j.cma.2015.09.013

56. Quinlan N, Lobovský L (2018) The finite volume particle method. Numerical methods and advanced simulation in biomechanics and biological processes. Elsevier, Amsterdam, pp 341–354

57. Quinlan NJ, Lobovský L, Nestor RM (2014) Development of the meshless finite volume particle method with exact and efficient calculation of interparticle area. Comput Phys Commun 85(6):1554–1563. https://doi.org/10.1016/j.cpc.2014.02.017

58. Oñate E, Idelsohn S, Zienkiewicz OC, Taylor RL, Sacco C (1996) A stabilized finite point method for analysis of fluid mechanics problems. Comput Methods Appl Mech Eng 139(1–4):315–346. https://doi.org/10.1016/s0045-7825(96)01088-2

59. Oñate E (1998) Derivation of stabilized equations for numerical solution of advective-diffusive transport and fluid flow problems. Comput Methods Appl Mech Eng 151(1–2):233–265. https://doi.org/10.1016/s0045-7825(97)00119-9

60. Löhner R, Sacco C, Oñate E, Idelsohn S (2001) A finite point method for compressible flow. Int J Numer Methods Eng 53(8):1765–1779. https://doi.org/10.1002/nme.334

Comput Part Mech 7(2):393–433. https://doi.org/10.1007/s40571-019-00272-x

61. Ortega E, Oñate E, Idelsohn S (2009) A finite point method for adaptive three-dimensional compressible flow calculations. Int J Numer Methods Fluids 60(9):937–971. https://doi.org/10.1002/fld.1892

62. Bajko J, Čermák L, Jícha M (2014) High order finite point method for the solution to the sound propagation problems. Comput Methods Appl Mech Eng 280:157–175. https://doi.org/10.1016/j.cma.2014.07.022

63. Liu MB, Xie WP, Liu GR (2005) Modeling incompressible flows using a finite particle method. Appl Math Model 29(12):1252–1270. https://doi.org/10.1016/j.apm.2005.05.003