# Proceedings

## of the

# XXVI Congreso de Ecuaciones Diferenciales y Aplicaciones
# XVI Congreso de Matemática Aplicada

**Gijón (Asturias), Spain**

**June 14-18, 2021**


SēMA
Sociedad Española
de Matemática Aplicada

Universidad de Oviedo

**Editors:**
Rafael Gallego, Mariano Mateos




Universidad de Oviedo
Servicio de Publicaciones de la Universidad de Oviedo
Campus de Humanidades. Edificio de Servicios. 33011 Oviedo (Asturias)
Tel. 985 10 95 03 Fax 985 10 95 07
http: www.uniovi.es/publicaciones
servipub@uniovi.es

ISBN: 978-84-18482-21-2

## Foreword

It is with great pleasure that we present the Proceedings of the 26$^{th}$ Congress of Differential Equations and Applications / 16$^{th}$ Congress of Applied Mathematics (XXVI CEDYA / XVI CMA), the biennial congress of the Spanish Society of Applied Mathematics S$\vec{e}$MA, which is held in Gijón, Spain from June 14 to June 18, 2021.

In this volume we gather the short papers sent by some of the almost three hundred and twenty communications presented in the conference. Abstracts of all those communications can be found in the abstract book of the congress. Moreover, full papers by invited lecturers will shortly appear in a special issue of the S$\vec{e}$MA Journal.

The first CEDYA was celebrated in 1978 in Madrid, and the first joint CEDYA / CMA took place in Málaga in 1989. Our congress focuses on different fields of applied mathematics: Dynamical Systems and Ordinary Differential Equations, Partial Differential Equations, Numerical Analysis and Simulation, Numerical Linear Algebra, Optimal Control and Inverse Problems and Applications of Mathematics to Industry, Social Sciences, and Biology. Communications in other related topics such as Scientific Computation, Approximation Theory, Discrete Mathematics and Mathematical Education are also common.

For the last few editions, the congress has been structured in mini-symposia. In Gijón, we will have eighteen minis-symposia, proposed by different researchers and groups, and also five thematic sessions organized by the local organizing committee to distribute the individual contributions. We will also have a poster session and ten invited lectures. Among all the mini-symposia, we want to highlight the one dedicated to the memory of our colleague Francisco Javier "Pancho" Sayas, which gathers two plenary lectures, thirty-six talks, and more than forty invited people that have expressed their wish to pay tribute to his figure and work.

This edition has been deeply marked by the COVID-19 pandemic. First scheduled for June 2020, we had to postpone it one year, and move to a hybrid format. Roughly half of the participants attended the conference online, while the other half came to Gijón. Taking a normal conference and moving to a hybrid format in one year has meant a lot of efforts from all the parties involved. Not only did we, as organizing committee, see how much of the work already done had to be undone and redone in a different way, but also the administration staff, the scientific committee, the mini-symposia organizers, and many of the contributors had to work overtime for the change.

Just to name a few of the problems that all of us faced: some of the already accepted mini-symposia and contributed talks had to be withdrawn for different reasons (mainly because of the lack of flexibility of the funding agencies); it became quite clear since the very first moment that, no matter how well things evolved, it would be nearly impossible for most international participants to come to Gijón; reservations with the hotels and contracts with the suppliers had to be cancelled; and there was a lot of uncertainty, and even anxiety could be said, until we were able to confirm that the face-to-face part of the congress could take place as planned.

On the other hand, in the new open call for scientific proposals, we had a nice surprise: many people that would have not been able to participate in the original congress were sending new ideas for mini-symposia, individual contributions and posters. This meant that the total number of communications was about twenty percent greater than the original one, with most of the new contributions sent by students.

There were almost one hundred and twenty students registered for this CEDYA / CMA. The hybrid format allows students to participate at very low expense for their funding agencies, and this gives them the opportunity to attend different conferences and get more merits. But this, which can be seen as an advantage, makes it harder for them to obtain a full conference experience. Alfréd Rényi said: "a mathematician is a device for turning coffee into theorems". Experience has taught us that a congress is the best place for a mathematician to have a lot of coffee. And coffee cannot be served online.

In Gijón, June 4, 2021

The Local Organizing Committee from the Universidad de Oviedo

## Scientific Committee

- Juan Luis Vázquez, Universidad Autónoma de Madrid
- María Paz Calvo, Universidad de Valladolid
- Laura Grigori, INRIA Paris
- José Antonio Langa, Universidad de Sevilla
- Mikel Lezaun, Euskal Herriko Unibersitatea
- Peter Monk, University of Delaware
- Ira Neitzel, Universität Bonn
- JoséÁngel Rodríguez, Universidad de Oviedo
- Fernando de Terán, Universidad Carlos III de Madrid

## Sponsors

- Sociedad Española de Matemática Aplicada
- Departamento de Matemáticas de la Universidad de Oviedo
- Escuela Politécnica de Ingeniería de Gijón
- Gijón Convention Bureau
- Ayuntamiento de Gijón

## Local Organizing Committee from the Universidad de Oviedo

- Pedro Alonso Velázquez
- Rafael Gallego
- Mariano Mateos
- Omar Menéndez
- Virginia Selgas
- Marisa Serrano
- Jesús Suárez Pérez del Río

# Contents

CONTENTS

XXVI Congreso de Ecuaciones Diferenciales y Aplicaciones
XVI Congreso de Matemática Aplicada
*Gijón, 14-18 junio 2021*
(pp. 228–235)

# An algorithm to create conservative Galerkin projection between meshes

**P. Gómez-Molina[1], L. Sanz-Lorenzo[2], J. Carpio[1]**
*1. Departamento de Ingeniería Energética*
*2. Departamento de Matemática Aplicada a la Ingeniería Industrial*
*E.T.S.I. Industriales, José Gutiérrez Abascal, 2, 28006 Madrid*
*Universidad Politécnica de Madrid*

**Abstract**

We present in this paper an algorithm to solve pure-convection problems with a conservative Lagrange-Galerkin formulation in the framework of the finite element method. The integrals obtained from the Lagrange-Galerkin formulation will be computed with an algorithm which leads to conservation of mass up to machine accuracy, when we transfer information from the mesh moved by the characteristic curves of the convection operator to the current mesh. The algorithm to compute the integrals considers the intersection of meshes composed by triangles (2-dimensions) and tetrahedra (3-dimensions) with straight sides. We will illustrate the good features of the method in terms of stability, accuracy and mass conservations in different pure-convection tests with non-divergence-free velocity fields.

## 1. Introduction

Nowadays, in the resolution of problems related with fluids, such as aerodynamics, combustion and heat transfer, we usually find convection-dominated equations. However, its resolution via finite element methods is not straightforward, since the treatment of the convective terms is a source of numerical problems due to the fact that the standard Galerkin formulation is unstable.

One methodology that brings about the stabilization of the convective term in a natural way is related to the Lagrangian description of the flow. Here, we use the information of the characteristic curves of the convection operator in order to integrate the equation in time. In the so-called Lagrange-Galerkin method (also known as Characteristic-Galerkin or semi-Lagrange-Galerkin method), see [1, 2], we identify each domain point $\mathbf{x}$ as a fluid particle at time $t_n$ and seek backward in time the position of this particle at time $t_{n-1}$, that we call the foot of the characteristic curve $\mathbf{X}(\mathbf{x}, t_n; t_{n-1})$, where the numerical solution $u_h(\mathbf{x}, t_{n-1}))$ is known. The set of the feet of the characteristic curves defines a backwards convected mesh, and the weak formulation of the problem performs a $L^2$-projection of the known solution from this convected mesh to the fixed mesh.

In the context of Lagrange-Galerkin schemes, Colera et al. [3] derived a conservative Lagrange-Galerkin formulation to solve pure-convection and convection-diffusion equations in the case of non-divergence-free velocity fields. The method is mainly based on formulating a conservation integral equation for a weighted mass, that can be discretized in time and in space with any order of accuracy, and is posed so that the terms that appear in the formulation can be easily computed by means of standard finite element operations.

Although the weak formulation in [3] leads to mass conservation, the right-hand side of the equation consists on an integral of functions that are defined in different element spaces (associated with the current triangulation and its backwards convected mesh). In [3], the integrals are computed with high-order quadrature rules [4], which is the reason the method proposed there is named "nearly-conservative". Since the basis functions of the fixed mesh are not polynomials, but only piecewise polynomials, over the elements of the convected mesh, the use of quadrature rules over such elements does not produce an exact result.

In this work, we propose an algorithm based on an appropriate mesh intersection procedure to accurately compute the right-hand side integral of the weak formulation. This techniques leads to better accuracy in the mass conservation property and also it improves the stability properties of the Lagrange-Galerkin scheme. Following Farrell et al. [5] we call this technique "supermesh technique".

The layout of the paper is as follows: Section 2 starts with the presentation of the conservative Lagrange-Galerkin formulation of pure convection problems with non-divergence-free velocity fields and concludes with the weak formulation of the problem in the framework of the finite element method. Section 3 constitutes the core of the paper, where we explain the numerical procedure to implement the "supermesh technique" to compute the integral of the right hand side term of the weak formulation of the problem, to transfer information from the moved mesh to the current mesh. In Section 4 we present numerical results to show the good properties of the proposed algorithm, in terms of stability, accuracy and mass conservation. Finally, some conclusions and comments are collected in Section 5.

## 2. Conservative Lagrange-Galerkin formulation for pure-convection problems

Let us consider the conservative form of a pure-convection equation for the scalar variable $u = u(\mathbf{x}, t)$

$$\begin{cases} \dfrac{\partial u}{\partial t} + \nabla \cdot (\mathbf{a}u) = 0 \quad \text{in } \Omega \times (0, T] \\ u(\mathbf{x}, 0) = u^0(\mathbf{x}) \quad \text{in } \Omega, \end{cases} \tag{2.1}$$

with $\Omega \subset \mathbb{R}^d$ (with $d = 2, 3$) a bounded domain with smooth boundary $\partial\Omega$, and $\mathbf{a}(\mathbf{x}, t)$ a regular velocity field with possible non-null divergence (we do not assume incompressible velocity). To solve numerically the problem (2.1) by means of a Lagrange-Galerkin scheme we have to divide the time interval $\overline{I} = [0, T]$ with a constant step size $\Delta t = t_n - t_{n-1}$. Associated with the velocity field $\mathbf{a}(\mathbf{x}, t)$ we can define the characteristic curves $\mathbf{X}(\mathbf{x}, t_n; t)$ of the convective or material derivative operator $D/Dt = \partial/\partial t + \mathbf{a} \cdot \nabla$ that correspond to the position backward in time of a fluid particle at time $t \leq t_n$ that will reach the domain point $\mathbf{x}$ at instant of time $t_n$. $\mathbf{X}(\mathbf{x}, t_n; t)$ is the solution to the system of equations

$$\begin{cases} \dfrac{d\mathbf{X}(\mathbf{x}, t_n; t)}{\partial t} = \mathbf{a}\left(\mathbf{X}(\mathbf{x}, t_n; t), t\right) \quad t < t_n, \\ \mathbf{X}(\mathbf{x}, t_n; t_n) = \mathbf{x}. \end{cases} \tag{2.2}$$

To obtain the weak conservative Lagrangian-Galerkin formulation of problem (2.1) we multiply the equation by a test function $v = v(\mathbf{x}, t)$ that satisfies the equation $Dv/Dt = 0$. Then, we obtain the expression

$$\frac{\partial (uv)}{\partial t} + \nabla \cdot (\mathbf{a}uv) = 0. \tag{2.3}$$

Now, we integrate (2.3) in the domain $\widetilde{\Omega}(t)$, that evolves backward in time from $t_n$ to $t < t_n$ according to the velocity field $\mathbf{a}(\mathbf{x}, t)$ and it is defined by the family of characteristic curves $\mathbf{X}(\mathbf{x}, t_n; t)$

$$\widetilde{\Omega}(t) := \left\{ \mathbf{X} \in \mathbb{R}^d : \ \mathbf{X} = \mathbf{X}(\mathbf{x}, t_n; t), \ \mathbf{x} \in \Omega \right\}$$

with $\widetilde{\Omega}(t_n) = \Omega$. Therefore, applying the Gauss theorem to the second term followed by the Reynolds theorem, we obtain the weak formulation of equation (2.3) as a temporal derivative over a integral extended to the fluid volume $\widetilde{\Omega}(t)$. For all test functions we have

$$\frac{d}{dt} \int_{\widetilde{\Omega}(t)} uv dX = 0 \quad \rightarrow \quad \int_{\Omega} u^n(\mathbf{x}) v^n(\mathbf{x}) dx = \int_{\widetilde{\Omega}(t_{n-1})} u^{n-1}(\mathbf{X}) v^{n-1}(\mathbf{X}) dX, \tag{2.4}$$

### 2.1. Finite element discretizacion and convected finite element space

The equation (2.4) forces us to consider an integration domain $\widetilde{\Omega}(t)$ that moves with the fluid particles, as well as test functions $v(\mathbf{x}, t)$ that remain constant along the fluid trajectories. Then, we chose as integration domain at instant of time $t_n$ the set $\Omega_h := \widetilde{\Omega}(t_n)$, a polygonal domain that approximates $\Omega$ and over which we define a regular triangulation $\mathbb{T}_h$ composed of triangles (in 2D) or tetrahedra (in 3D).

Associated with the triangulation $\mathbb{T}_h$ we define a conforming finite element space $V_h$ where the numerical solution $u_h^n(\mathbf{x})$ (shorthand for $u_h(\mathbf{x}, t_n)$) is computed. To do so, we consider a reference element $\hat{K} \in \mathbb{R}^d$ and define $\hat{V}_h$ as the space $P_m$ of polynomial functions of degree less or equal to $m$ and denote it dimension by $n_v$. For each element $K \in \mathbb{T}_h$ we define the one-to-one affine mapping $F_K : \hat{K} \rightarrow K$

$$\mathbf{F}_K : \hat{K} \longrightarrow K, \ \mathbf{x} = J_K \hat{\mathbf{x}} + \mathbf{b}_K, \quad J_K \in \mathbb{R}^{d \times d} \text{ and } \mathbf{b}_K \in \mathbb{R}^d. \tag{2.5}$$

and we denote $V_h$ the resulting conforming finite element space and $N_v$ the number of mesh nodes.

Now, each element $K \in \mathbb{T}_h$ which composes domain $\Omega_h$ is convected backwards in time from $t_n$ to $t_{n-1}$ obtaining $\widetilde{K}^{n-1}$ as the geometric place of the so-called feet of the characteristic curves $\mathbf{X}(\mathbf{x}, t_n; t_{n-1})$ with $\mathbf{x} \in K$, for which we use the shorthand $\mathbf{X}^{n-1}(\mathbf{x})$. We are going to consider an approximation $\widetilde{K}_h^{n-1}$ of $\widetilde{K}^{n-1}$ given by the following isoparametric transformation

$$\widetilde{\mathbf{F}}_K^{n-1} : \widehat{\mathbf{x}} \in \hat{K} \longrightarrow \mathbf{X}_h \in \widetilde{K}_h^{n-1}, \quad \mathbf{X}_h^{n-1}(\hat{\mathbf{x}}) = \sum_{i=1}^{n_v} \mathbf{X}^{n-1}(\mathbf{v}_i) \hat{\varphi}_i(\hat{\mathbf{x}}), \tag{2.6}$$

with $\mathbf{v}_i$ the coordinates of the $i$-th local node of $K$ and $\hat{\varphi}_i$ the $i$-th elemental nodal basis function of $\hat{V}_h$. The transformation $\widetilde{\mathbf{F}}_K^{n-1}$ incurs an error in the approximation of $\mathbf{X}^{n-1}(\mathbf{x})$ consistent with the space finite element discretization, i.e.,

$$|\mathbf{X}^{n-1}(\mathbf{x}) - \mathbf{X}_h^{n-1}(\mathbf{x})| = O(h^{m+1}),$$

Note that $\mathbf{X}^{n-1}(\mathbf{v}_i)$ must be computed for the $N_v$ mesh nodes $\{\mathbf{v}_i\}_{i=1}^{N_v}$ solving numerically the differential equation (2.2). Moreover, the transformation $\widetilde{\mathbf{F}}_K^{n-1}$ has the advantage that $\mathbf{X}_h^{n-1}(\mathbf{x})$ and $\mathbf{x}$ share the same natural coordinates in the reference element [1, 6], that is,

$$\mathbf{X}^{n-1}(\mathbf{x}) \simeq \mathbf{X}_h^{n-1}(\mathbf{x}) := \widetilde{\mathbf{F}}_K^{n-1}\left(\mathbf{F}_K^{-1}(\mathbf{x})\right). \tag{2.7}$$

For each $K \in \mathbb{T}_h$, the definition of the isoparametric transformation $\widetilde{\mathbf{F}}_K^{n-1}$ of (2.6) leads to an element $\widetilde{K}_h^{n-1}$ with polynomial edges of degree $m$, the order of the finite element space approximation $\hat{V}_h$, and which is an aproximation of the real convected element $\widetilde{K}^{n-1}$. Let $\widetilde{\mathbb{T}}_h^{n-1}$ be the mesh composed by these approximate convected elements $\widetilde{\mathbb{T}}_h^{n-1} := \left\{\widetilde{K}_h^{n-1} : K \in \mathbb{T}_h\right\}$. In the present paper we are going to consider linear finite elements, $m = 1$, in order to have approximated convected elements $\widetilde{K}_h^{n-1}$ with straight edges (sides in 3D). We can see Fig. 1 for an explanatory scheme of the above explanation.

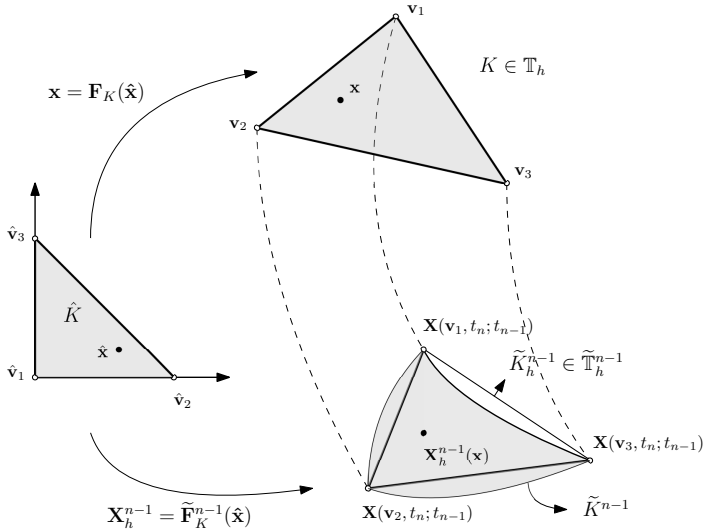


**Fig. 1** Convected nodes $\mathbf{X}^{n-1}(\mathbf{v}_i)$ backward in time from triangle $K \in \mathbb{T}_h$. From the convected nodes, we approximate the real convected element by an isoparametric $\widetilde{K}_h^{n-1} \in \widetilde{\mathbb{T}}_h^{n-1}$ element via application $\mathbf{X}_h^{n-1}(\mathbf{x}) = \widetilde{\mathbf{F}}_K^{n-1}\left(\mathbf{F}_K^{-1}(\mathbf{x})\right)$.

Now, let us consider a function $v_h(\mathbf{x}, t)$ which verifies $Dv_h/Dt = 0$ and $v_h^n(\mathbf{x}) \in V_h$. Since $v_h$ is constant along the fluid trajectories, we can make the approximation $v_h^{n-1}\left(\mathbf{X}_h^{n-1}(\mathbf{x})\right) = v_h^n(\mathbf{x})$. Moreover, $v_h^{n-1}$ belongs to the convected finite element space

$$\widetilde{V}_h^{n-1} := \left\{v_h^{n-1} : v_h^{n-1}\left(\mathbf{X}_h^{n-1}(\mathbf{x})\right) = v_h^n(\mathbf{x}) \in V_h\right\},$$

and hence $\widetilde{V}_h^{n-1}$ is also a $P_m$ space, but associated to the mesh $\widetilde{\mathbb{T}}_h^{n-1}$ instead of $\mathbb{T}_h$.

Finally, with the definition of $\Omega_h$, $\widetilde{\Omega}_h^{n-1}$ and their associated triangulations $\mathbb{T}_h$, $\widetilde{\mathbb{T}}_h^{n-1}$ and the finite element spaces $V_h$, $\widetilde{V}_h^{n-1}$, we can define the numerical approximation of the weak conservative formulation (2.4) via the finite element method

$$\int_{\Omega_h} u_h^n v_h^n dx = \int_{\widetilde{\Omega}_h^{n-1}} u_h^{n-1}(\mathbf{X}) v_h^{n-1}(\mathbf{X}) dX, \quad \forall v_h^n \in V_h \ \text{ (and its associated function } v_h^{n-1} \in \widetilde{V}_h^{n-1}), \tag{2.8}$$

where the initial value $u_h^0$ is taken as the $L^2$ projection of the initial condition $u^0(\mathbf{x})$, i.e.,

$$\int_{\Omega_h} u_h^0 v_h^0 dx = \int_{\Omega_h} u^0 v_h^0 dx, \quad \forall v_h^0 \in V_h. \tag{2.9}$$

The computation of the left-hand side of (2.8) lead us to the standard mass matrix associated to the triangulation $\mathbb{T}_h$ and it is straightforward to compute it, however the right-hand side in (2.8) involves the product of $u_h^{n-1}$, which is defined piecewise in $\mathbb{T}_h$, and $v_h^{n-1}$, which is defined piecewise in $\widetilde{\mathbb{T}}_h^{n-1}$ (see Fig. 2). Since these two meshes are usually different, to compute this right-hand side we can follow the following strategies:

1) Integrate over the elements in $\widetilde{\mathbb{T}}_h$ with high-order quadrature rules [4] as is done in Colera et al. [3].

2) Develop a mesh intersection technique to accurately compute this term [5, 7, 8], which leads to better accuracy in mass conservation. In the present work we adopt this approach and derive a technique to develop mesh intersection of straight triangles in 2D and straight tetrahedra in 3D of high efficiency using very conventional operations within the finite element methodology.

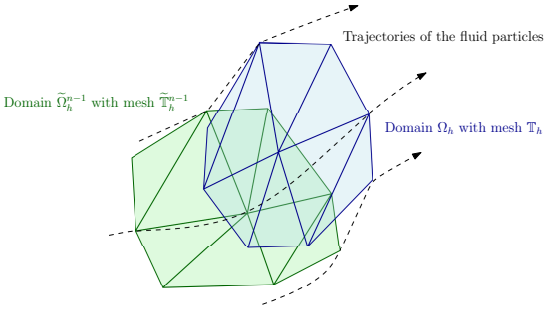


**Fig. 2** Scheme that illustrates the domains and meshes that appear in the formulation. The nodes of the elements in $\mathbb{T}_h$ are convected backwards in time with the flow velocity field to form $\widetilde{\mathbb{T}}_h^{n-1}$. The variables $u_h^n$, $v_h^n$ and $u_h^{n-1}$ are defined piecewise in $\mathbb{T}_h$, whereas $v_h^{n-1}$ is defined piecewise in $\widetilde{\mathbb{T}}_h^{n-1}$.

Note that, if in problem (2.1) the velocity field satisfies $\mathbf{a} \cdot \mathbf{n} = \mathbf{0}$ on the boundary $\partial\Omega_h$, with $\mathbf{n}$ the outward normal vector on the boundary, then the backward convected domain will be the same as the current volume (fluid particles on the boundary $\partial\Omega_h$ do not cross the boundary), i.e., $\Omega_h = \tilde{\Omega}_h^{n-1}$ (but $\mathbb{T}_h \neq \widetilde{\mathbb{T}}_h^{n-1}$ if $\mathbf{a} \neq \mathbf{0}$). That means that the conservation principle is satisfied in the domain $\Omega_h$ for all instants of time

$$\int_{\Omega_h} u_h^n(\mathbf{x})dx = \int_{\Omega_h} u_h^0(\mathbf{x})dx, \quad \forall t_n \in [0, T]. \tag{2.10}$$

## 3. Algorithm for the intersection of meshes with straight elements

To compute (2.8) numerically, we replace the test function $v_h^n$ by each of the $N_v$ basis functions $\varphi_I(\mathbf{x}) \in V_h$ and $u_h^n = \sum_{J=1}^{N_v} U_J^n \varphi_J(\mathbf{x})$. Then, the left hand side reads

$$\int_{\Omega_h} u_h^n(\mathbf{x})\varphi_I(\mathbf{x})dx = \sum_{K \in \mathbb{T}_h} \left( \sum_{J=1}^{N_v} U_J^n \int_K \varphi_I(\mathbf{x})\varphi_J(\mathbf{x})dx \right),$$

which can be computed exactly with a quadrature rule since both $\varphi_I(\mathbf{x})$ and $\varphi_J(\mathbf{x})$ are basic function in $V_h$. However, for the right hand side we have

$$\int_{\widetilde{\Omega}_h^{n-1}} u_h^{n-1}(\mathbf{X}^{n-1}(\mathbf{x}))\varphi_I^{n-1}(\mathbf{X}^{n-1}(\mathbf{x}))dX = \sum_{\widetilde{K}_h^{n-1} \in \widetilde{\mathbb{T}}_h^{n-1}} \left( \sum_{J=1}^{N_v} U_J^{n-1} \int_{\widetilde{K}_h^{n-1}} \varphi_I^{n-1}(\mathbf{X}^{n-1}(\mathbf{x}))\varphi_J(\mathbf{X}^{n-1}(\mathbf{x}))dX \right), \tag{3.1}$$

and inside the integral there are two kinds of basis functions: $\varphi_J \in V_h$ associated with the elements $K \in \mathbb{T}_h$ and $\varphi_I^{n-1} \in \widetilde{V}_h^{n-1}$ associated with the elements $\widetilde{K}_h^{n-1} \in \widetilde{\mathbb{T}}_h^{n-1}$. Therefore, as the basis functions are piecewise polynomials over each element of their respective meshes, the use of intersection techniques is required to compute this right hand side exactly with quadrature rules, as can be seen in Fig. 2.

In this work we propose a mesh intersection algorithm based in [5]. We are going to simplify the notation in accordance with those introduced in that paper. We define $\mathbb{T}_h$ as the donor mesh, and denote donor elements $K_D \equiv K \in \mathbb{T}_h$, whereas we define as $\widetilde{\mathbb{T}}_h^{n-1}$ the target mesh, so that the target elements are $K_T \equiv \widetilde{K}_h^{n-1} \in \widetilde{\mathbb{T}}_h^{n-1}$. Moreover, as we have considered in Fig. 1 and Fig. 2, we are going to illustrate the main stages of the algorithm with figures of meshes composed of triangles in two-dimensions. For tetrahedra in three-dimensions the situation is analogous, but spatial figures are more difficult to understand. Then, the algorithm to compute integral (3.1) has the following stages:

1. For each $K \in \mathbb{T}_h$, find the set of elements $K_D^T := \{K_{D_1}, K_{D_2}, ...\} \subset \mathbb{T}_h$ such that $K_{Di} \bigcap K_T \neq \emptyset$ with $K_T \in \widetilde{\mathbb{T}}_h^{n-1}$. Those elements are shown in Fig. 3.

2. Compute the supermesh associated with the intersection $K_{Di} \bigcap K_T$ for each $K_{D_i} \in K_D^T$. The supermesh is the set $K_{TD_i} = \{K_{TD_i,1}, K_{TD_i,2}, ...\}$ where $K_{TD_i}$ is included in both $K_T$ and $K_{D_i}$. Fig. 4 shows a scheme of this stage of the algorithm. To create a triangulation of the intersection zone there are several procedures, and one of the simplest (useful for convex polytopes in two and three dimension) is the Sutherland-Hodgman clipping algorithm [9].

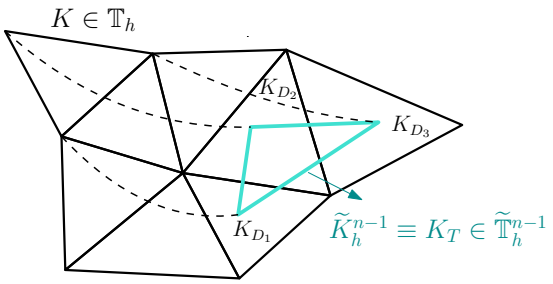


**Fig. 3** The element $K_T \in \widetilde{\mathbb{T}}_h^{n-1}$ (convected from $K \in \mathbb{T}_h$) intersects with the triangles of the set $K_D^T = \{K_{D_1}, K_{D_2}, K_{D_3}\} \in \mathbb{T}_h$.

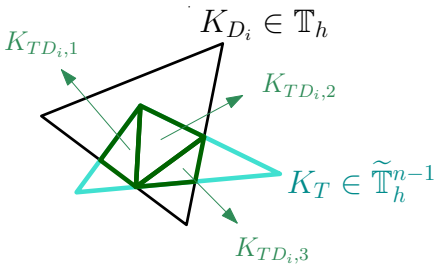


**Fig. 4** The intersection of $K_T$ and $K_{D_i}$ is remeshed to obtain the supermesh. In this example that supermesh has three elements: $K_{TD_i,1}$, $K_{TD_i,2}$ and $K_{TD_i,3}$

3. Define the linear transformations $\hat{\mathbf{x}}_{D_i} = \mathbf{g}_{D_iT,j}(\hat{\mathbf{x}})$ and $\hat{\mathbf{x}}_T = \mathbf{g}_{TD_i,j}(\hat{\mathbf{x}})$ between a common reference element and a reference element associated to $K_{D_i}$ and $K_T$, respectively. This is shown in Fig. 5

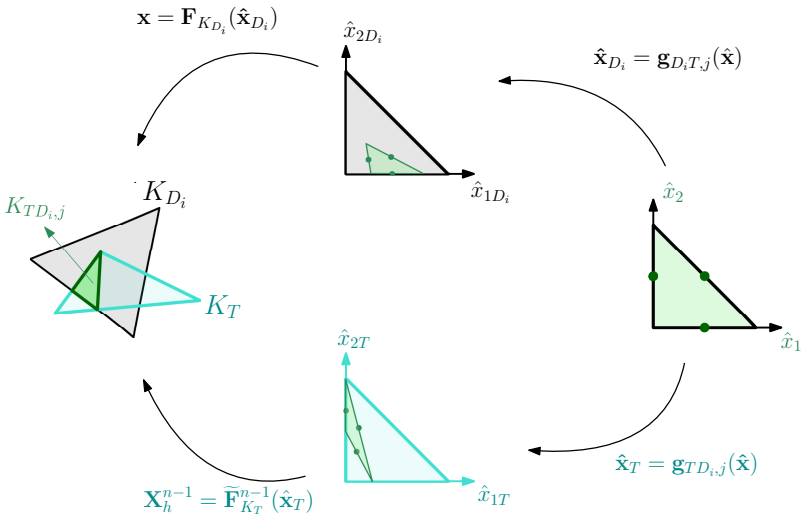


**Fig. 5** Linear transformations from the standard reference element $\hat{K}$ to those reference elements $\hat{K}_{D_i}$ and $\hat{K}_T$ associated with the proper elements $K_{D_i}$ and $K_T$, respectively.

4. Finally, the integral $m_{IJ}|_{K_T} := \int_{\widetilde{K}_h^{n-1}} \varphi_I^{n-1}\left(\mathbf{X}^{n-1}(\mathbf{x})\right) \varphi_J\left(\mathbf{X}^{n-1}(\mathbf{x})\right) dX$ can be computed as:

$$m_{IJ}|_{K_T} = \sum_{K_{D_i} \in K_D^T} \left[ \sum_{K_{TD_i,j} \in K_{TD_i}} \int_{K_{TD_i,j}} \varphi_I^{n-1}\left(\mathbf{X}^{n-1}(\mathbf{x})\right) \varphi_J\left(\mathbf{X}^{n-1}(\mathbf{x})\right) dX \right]$$

with

$$\int_{K_{TD_i,j}} \varphi_I^{n-1}\left(\mathbf{X}^{n-1}(\mathbf{x})\right) \varphi_J\left(\mathbf{X}^{n-1}(\mathbf{x})\right) dX = \sum_{s=1}^{n_q} \omega_s \hat{\varphi}_I\left(\mathbf{g}_{TD_i,j}(\hat{\boldsymbol{\xi}}_s)\right) \hat{\varphi}_J\left(\mathbf{g}_{D_iT,j}(\hat{\boldsymbol{\xi}}_s)\right) \det\left(\frac{\partial \widetilde{\mathbf{F}}_{K_T}(\hat{\mathbf{x}}_T)}{\partial \hat{\mathbf{x}}_T}\right) \det\left(\frac{\partial \mathbf{g}_{TD_i,j}(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}}\right)$$

where $\hat{\boldsymbol{\xi}}_s$, $s = 1, ..., n_q$ are the quadrature points in the reference element and the product $\det\left(\frac{\partial \widetilde{\mathbf{F}}_{K_T}(\hat{\mathbf{x}}_T)}{\partial \hat{\mathbf{x}}_T}\right) \det\left(\frac{\partial \mathbf{g}_{T D_i,j}(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}}\right)$ equals the size of the supermesh element, $\left|K_{T D_i,j}\right|$

## 4. Numerical test

Next, we present a numerical test to illustrate the performance of the numerical algorithm. Note that the trajectories of the fluid particles can be computed with high accuracy or analytically for these problem, which allows us to obtain the exact solution of pure-convection problems (2.1) through the formula $u(\mathbf{x}, T) = u^0\left(\mathbf{X}(\mathbf{x}, T; 0)\right) \det\left(\frac{\partial \mathbf{X}(\mathbf{x}, T; 0)}{\partial \mathbf{x}}\right)$ [10], which means

$$u(\mathbf{x}, T) = u^0(\mathbf{X}(\mathbf{x}, T; 0)) \exp\left(-\int_0^T \left[(\nabla \cdot \mathbf{a})\right]_{\mathbf{X}(\mathbf{x},T;t)} dt\right).$$

We are going to show numerical results for the projection technique presented in this paper via intersection of meshes. This technique is called "supermesh projection" as opposed to the "standard projection" where we use high-order quadrature rules to compute the integrals with high accuracy. For the supermesh projection we need quadrature rules of order only two (both in 2-dimension ($n_q = 3$) and 3-dimension ($n_q = 4$)), the minimum order needed to integrate exactly the corresponding product of basis functions. For the standard projection we show in figures the number of quadrature points used in each numerical simulation.

The test consists of a pure convection problem (2.1) with the following velocity field and initial condition:

$$\mathbf{a}(\mathbf{x}, t) = \left[0.45 + \sin(t - x_1),\ 0.45 + \sin(t - x_2)\right]^{\mathrm{T}}, \quad \text{and} \quad u^0(\mathbf{x}) = \exp\left(-200\left(2 - \cos(x_1) - \cos(x_2)\right)\right).$$

in a domain $\Omega_h = [-1, 1] \times [-1, 1]$ and final instant of time T=0.5. In this problem $\mathbf{a} \cdot \mathbf{n} \neq \mathbf{0}$, but the solution on the boundary for all instant of time $t \leq T$ is negligible $u|_{\partial\Omega} \simeq 0$ and then the mass of the solution in the domain $\Omega_h$ is almost maintained (2.10).

The evolution of the solution with time can be seen in Fig.6. At time $t = 0$ we have a gaussian hill in the middle of the domain. When the time goes on, the solution moves along the line $y = x$ and its width is reduced and the value of its vertex is increased, to satisfy the conservation of mass at all instant of times following (2.10).

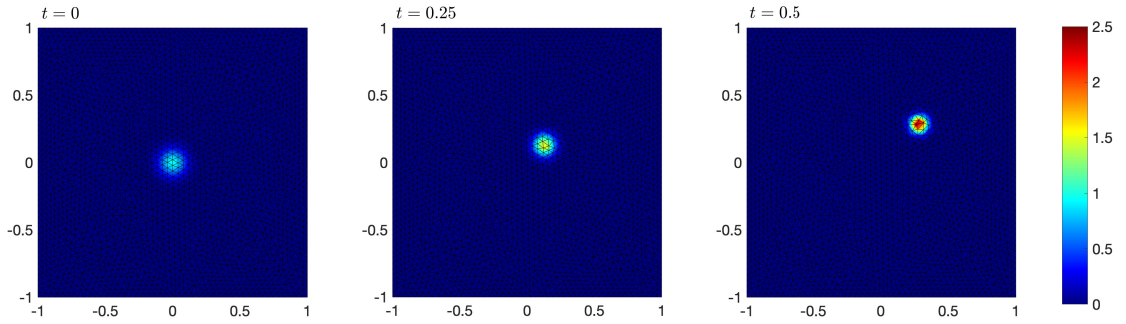


**Fig. 6** Numerical solution of test at different instants of time in a uniform triangulation $\mathbb{T}_h$ with $h = 0.085$.

Now, we are going to measure the error in the $L^2$-norm between the numerical solution and the exact solution and also the mass error, both at the last instant of time. We consider a time step size $\Delta t = 0.005$ and different meshes composed of regular elements of size $h$. The results can be seen in Fig.7. The $L^2$ error for supermesh projection and standard projection is similar in both cases, and also its convergence with $h$ is close to the theoretical one shown in the black dashed line ($O(h^2)$ for linear finite elements). However, the difference between supermesh projection and standard projection is remarkable in terms of mass conservation. We can observe in Fig. 7 that supermesh projection is up to twelve orders of magnitude more accurate when computing the mass error.

Finally, in Fig.8 we show an interesting phenomenon when we analyze the $L^2$-error at the last instant of time, as a function of time step size $\Delta t$. Now, we are going to consider a uniform mesh with with $h = 0.085$ and make several numerical simulations with different time step sizes $\Delta t$. It is known that standard projection may get unstable when using small time steps if the number of quadrature points used for integration is not high enough. However, for the supermesh projection, this instability does not appear, since the integrals are computed exactly with low number of quadrature points (the minimum needed to integrate exactly the corresponding product between basis functions).

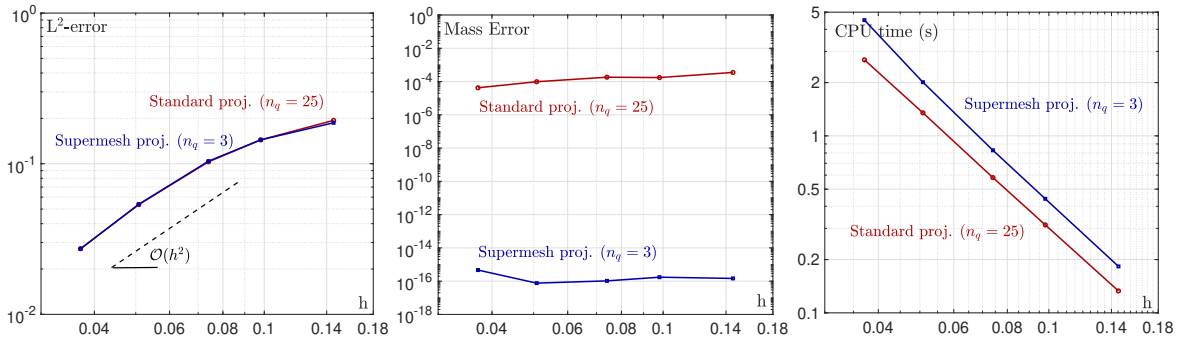


**Fig. 7** $L^2$-error (on the left), mass error (in the middle), and computational requirement in terms of CPU time (on the right) for different uniform meshes with size $h$ and $\Delta t = 0.005$ maintained constant in all the simulations.

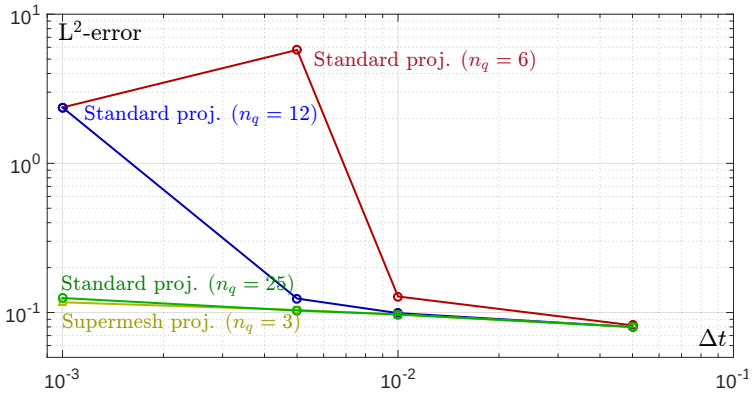


**Fig. 8** $L^2$-error for different time steps and the same mesh with $h = 0.085$.

## 5. Conclusion

In this work we described in detail an algorithm to compute exactly the integrals that appear in the conservative Lagrange-Galerkin formulation of a pure-convection problem in a linear finite element framework. This technique is called "supermesh projection" as opposed to the "standard projection" where we use high-order quadrature rules to compute the integrals with high accuracy. Both techniques have almost identical $L^2$-error with moderately large time step sizes $\Delta t$ and high quadrature rules for the standard projection (for supermesh projection we need quadrature rules of order two, both in 2- and 3-dimensions). However, the main advantage of supermesh projection is that it avoids instabilities when the integrals are computed with enough accuracy in standard projection and when numerical error are accumulated in time (small time step sizes $\Delta t$). The main disadvantage of the supermesh projection is that the computational requirement of the right hand side of the weak formulation is larger than standard projection (nearly a factor two when the number of element which conformed the mesh is large). However, in practice this issue is not a significant problem, since the Lagrange-Galerkin formulation usually is used in other more complicated problems as convection-diffusion-reaction equations or Navier-Stokes equations. In these cases, computing the convection terms is usually the less resource consuming step, so this time increase would not be that relevant for the overall process. Moreover, the intersection mesh procedure can be carried out with straightforward parallel programming. As future work, we want to extent the present algorithm of mesh intersection to finite elements of high order ($m > 1$ order). In this case, convected elements $\widetilde{K}_h^{n-1}$ have sides given by a polynomial function of degree $m$, and the intersection of elements is much more complicated.

## Acknowledgments

This research has been partially funded from the "Ministerio de Ciencia, Innovación y Universidades" of Spain and European Regional Development Funds by project PGC-2018-097565-BI00.

## References

[1] R. Bermejo, J. Carpio, A semi-Lagrangian-Galerkin projection scheme for convection equations, IMA Journal of Numerical Analysis, Vol. 30(3) (2010) 799-831.

[2] R. Bermejo, P. Galán del Sastre, L. Saavedra, *A second order in time modified Lagrange–Galerkin finite element method for the incompressible navier-stokes equations*, SIAM J. Numer. Anal. 50(6) (2012) 3084–3109.

[3] M. Colera, J. Carpio, R. Bermejo, A nearly conservative high order Lagrange Galerkin method for the resolution of scalar convection-dominated equations in non-divergence-free velocity fields, Computer Methods in Applied Mechanics and Engineering, 372(1) (2020) 113366.

[4] K.W. Morton, A. Priestley, E. Süli, Stability of the Lagrange-Galerkin method with non-exact integration, ESAIM: Math. Model. Numer. Anal. 22(4) (1988) 625-653.

[5] P. Farrell, J. Maddison, Conservative interpolation between volume meshes by local Galerkin projection, Comput. Methods Appl. Mech. Engrg., 200 (2011) 89-100.

[6] J. Carpio, J.L. Prieto, M. Vera, Local anisotropic adaptive algorithm for the solution of low-Mach transient combustion problems, J. Comput. Phys. 306 (2016) 19-42.

[7] F. Alauzet, A parallel matrix-free conservative solution interpolation on unstructured tetrahedral meshes, Comput. Methods Appl. Mech. Engrg., 299 (2016) 116-142.

[8] M. Tabata, S. Uchiumi, A genuinely stable Lagrange-Galerkin scheme for convection-diffusion problems, Japan J. Indust. Appl. Math, 33 (2016) 121-143.

[9] I.E. Sutherland, G.W. Hodgman, Reentrant polygon clipping, Commun. ACM 17(1) (1974) 32-42.

[10] R. Bermejo, J. Conde, A conservative Quasi-Monotone Semi-Lagrangian Scheme, Mon. Wea. Rev., 130 (2002) 423-430.