Lagrange-Galerkin methods for convection-diffusion equations

Rodolfo Bermejo
Depto. Matemática Aplicada, ETSII, Universidad Politécnica de Madrid, España

Abstract

In these notes we introduce and study Lagrange-Galerkin methods of order two. We present algorithms to implement them efficiently, and develop an error analysis that shows the different regimes of convergence of such methods.

1 Introduction

The design of efficient and accurate convection-diffusion algorithms is of significant importance in the computational fluid dynamics community, in particular, when the transport terms of the equations describing the mathematical model become dominating with respect to the diffusive ones. In this case there appear a large variety of spatio-temporal scales that have to be properly resolved in order to obtain a numerical solution sufficiently close to the exact one. To see this is so, we consider the prototype equation for the convection-diffusion problem of a passive substance, the concentration of which is denoted by $c(x,t)$, in a bounded domain $D \subset \mathbb{R}^d$ with smooth boundary Γ:

$$\begin{cases} \dfrac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c - \nu \Delta c = f, \quad \text{in } D \times (0,T) \\ c_{|\Gamma} = 0 \\ c(0) = c_0 \end{cases}$$

where $\mathbf{u}$ is the velocity and ν the diffusion coefficient. The dimensionless form of this equation contains the so-called Péclet number Pe defined as $Pe = \dfrac{UL}{\nu}$, where U and L represent characteristic velocity and length scales respectively. When Pe is large enough, two sources of difficulty appear in the numerical treatment of this problem. The first one arises from the fact that the diffusive term, $\nu\Delta c$, may be considered as a perturbation to the convective one, $\dfrac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c$, in regions where $c(x,t)$ is smooth; so that, in these regions the dynamics of the solution is mainly governed by $\frac{\partial c}{\partial t}+\mathbf{u}\cdot\nabla c$, which represents the change of c along the characteristic curves (or trajectories of the flow particles) of the hyperbolic operator $\frac{\partial}{\partial t}+\mathbf{u}\cdot\nabla$. But the existence of boundary conditions to be satisfied by $c(x,t)$ in all $\Gamma\times(0,T)$ is incompatible with the hyperbolic character of $\frac{\partial c}{\partial t}+\mathbf{u}\cdot\nabla c$; so that, the imposition of the boundary conditions on $c(x,t)$ will lead to the appearance of a region at the boundary where the solution has to accommodate to satisfy the boundary conditions. This region is termed as boundary layer, and one can show through perturbation analysis that the width of it is $O(Pe^{-\alpha})$, $0<\alpha<1$. Therefore, for high Péclet numbers the boundary layer is narrow and consequently the solution will develop a strong gradient inside it. It is well known, see for instance [21], that numerical methods based on Galerkin projection (either as finite elements, or as spectral methods, or as hp finite elements) will develop spurious oscillations (Gibbs phenomenon), unless the boundary layer is properly resolved, which will have a pernicious polluting effect on the solution. A primary remedy for this consists of allocating a large number of elements in the narrow boundary. This fact brings us to the second source that concerns with the size of the time step Δt one has to choose for conventional time discretizations schemes of the equation. At high Péclet numbers, time discretization explicit schemes are ruled out due to the fact that the stability criterion of these schemes requires Δt to be so small that the number of time steps needed to perform the simulation is very large, in particular, when T is long. Other frequently used time schemes are the so called semi-implicit schemes in which the diffusive terms are discretized implicitly and the convective terms are left explicit. This discretization yields linear symmetric systems of algebraic equations which are efficiently solved by the preconditioned conjugate gradient method or by multigrid algorithms; however, it is subjected to the stability criterion of the form

$$\frac{\Delta t \max\limits_{D\times(0,T)}|\mathbf{u}|}{h_{\min}}\leq K_1,\ 0<K_1<1,$$

where $|\mathbf{u}|$ denotes the modulus of the velocity vector and $h_{\min}$ is the minimum diameter of the mesh elements. Clearly, when Pe is large the size of Δt is also unreasonably small in semi-implicit schemes. Therefore, at high Péclet numbers Pe it is convenient to use implicit schemes which are unconditionally stable and do not link the size of Δt to $h_{\min}$; however, this requires the use of non-symmetric solvers which are less efficient than the solvers for symmetric systems.

Considering that the Navier-Stokes equations can, roughly speaking, be viewed as non-linear convection-diffusion equations, and given the industrial and scientific interest that flows at high Reynolds numbers Re have (in Navier-Stokes equations Re plays the role of Pe in convection-diffusion equations for a passive substance), many efforts have been devoted to the development of numerical algorithms in the framework of the Galerkin projection to overcome the drawbacks above described.

This development has followed different approaches, such as the Eulerian, the Lagrangian and the Eulerian-Lagrangian ones. In the Eulerian approach one calculates mesh-point values of c at time instants t_n, thus formulating the numerical method on a fixed mesh but with the purpose of suppressing the wiggles in an efficient manner without damaging the accuracy of the method. To this respect, we shall only refer to Petrov-Galerkin methods such the SUPG (Stream-Upwind-Petrov-Galerkin) and Galerkin/least squares algorithms developed by Hughes and coworkers [13], [19] for convection-diffusion problems of a passive substance as well as the Navier-Stokes equations and conservation laws. This is a general finite element method for problems with strong hyperbolic terms, including compressible and incompressible flows, which has been developed by introducing two modifications to the conventional Galerkin method. The first one consists of modifying the test functions along the streamlines of the flow to yield a least-square control of the residual of the finite element solution. The second modification adds an artificial diffusion term of strength $Ch_K^2R_K$, where h_K is the local mesh size and R_K is the local finite element residual in this method.

In the Lagrangian approach one attempts to devise a stable numerical method by allowing the mesh to follow the trajectories of the flow. The problem now is that the mesh undergoes large deformations, after a number of time steps, due to stretching and shearing, and consequently some sort of remeshing has to be done in order to proceed with the calculations. The latter may become a source of large errors.

In the Eulerian-Lagrangian approach the purpose is to get a method that combines the good properties of both the Eulerian and Lagrangian approaches. There have been various methods trying to do so, among them we shall cite the characteristics streamline diffusion (hereafter, CSD), the backwards Lagrange-Galerkin or simply Lagrange-Galerkin (LG) methods (also termed Characteristics Galerkin), the semi-Lagrangian methods, and more recently a new class of LG methods proposed by M. Benítez in her excellent PhD. thesis (2009) to integrate the convection-diffusion equations formulated in Lagrangian coordinates, thus in this notes we term this latter LG methods as forward Lagrange-Galerkin (FLG) methods to make clear that they integrate the equation forward in time along the trajectories of the flow. The CSD method has been developed by [17], [18] and [20] and intends to combine the good properties of both the Lagrangian methods and the streamline diffusion method by orienting the space-time mesh along the characteristics in space-time, yielding thus to a particular version of the streamline diffusion method. The Lagrange-Galerkin and semi-Lagrangian methods approximate the material derivative

$$\frac{Dc}{Dt} = \frac{\partial c}{\partial t} + \mathbf{u}\cdot\nabla c$$

at each time step by a backwards in time discretization along the characteristics trajectories $X(x, t_{n+1}; t)$ of the operator $\dfrac{\partial}{\partial t} + \mathbf{u}\cdot\nabla$, $t_{n-l} \leq t < t_{n+1}$, l being an integer that usually takes the values 0 or 1. At $t = t_{n+1}$, $X(x, t_{n+1}; t_{n+1}) = x \in \overline{D}$. The diffusion terms are implicitly discretized on the fixed mesh generated in $\overline{D}$. The point here is how to evaluate $c(X(x, t_{n+1}; t), t)$. One way is by L^2 projection onto the finite dimensional space associated with the fixed mesh, as the Lagrange-Galerkin method does, see [5], [6], [12], [23], [14] and [28] just to cite a few; another way is by polynomial interpolation projection of high order as [10] and [16] propose. When the evaluation of $c\left(X\left(x,t_{n+1};t\right),t\right)$ is done by polynomial interpolation projection the method is called semi-Lagrangian. The FLG methods of [3], formulated in a finite element framework, integrate the Lagrangian formulation of the convection-diffusion equation forward in time along the characteristics curves of the transport operator, followed by the Galerkin projection onto the finite dimensional space associated with the mesh defined in the space of material points or labels. The advantages of LG or semi-Lagrangian methods are various. From a practical point of view we have the following: (i) they allow a large time step without damaging the accuracy of the solution; (ii) unlike the pure Lagrangian methods, they do not suffer from mesh-deformation, so that no remeshing is needed; (iii) they yield algebraic symmetric systems of equations to be solved. From a numerical analysis point of view, we shall show in these notes that the constant C that appears in the error estimates of the LG methods is much smaller than the corresponding constant of the standard Galerkin methods and, what it is more important, is uniformly bounded with respect to the values of ν. To appreciate the relevance of this behavior of C, we recall [24] that the error constant of standard Galerkin methods in convection-diffusion problems takes the form $C_G \sim \nu^{-1}\exp(t\max_{D\times(0,T)}|\mathbf{u}|\,\nu^{-1})$, and is sharp because the Gibbs phenomenon, which appears in the boundary layers when the mesh is coarse, grows exponentially. The ν^{-1} dependence of the constant C_G makes the standard Galerkin methods be unreliable in convection dominated diffusion because in such problems $\nu \ll 1$ and therefore C_G becomes very large. This does not happen in LG methods because the dependence on ν is uniformly bounded; however, this does not mean that LG are free from the Gibbs phenomenon if the grid is coarse, but such a phenomenon is well under control and so is its pollutant effect. The semi-Lagrangian methods have been used in Meteorology for numerical weather prediction since the early 80's of the past century, see [27], and from then on they have become the scheme for some of the present generation of sophisticated weather prediction models such as HIRLAM. Recently, in [9] the semi-Lagrangian has been combined with second order implicit-explicit Runge-Kutta-Chebyshev schemes in a finite element framework to develop a time-space adaptive algorithm for convection reaction-diffusion problems.

PART I
LG method for linear convection equations

2 The continuum problem

To introduce the idea of the LG method we consider the Cauchy problem for a linear hyperbolic equation of first order

$$\begin{cases} \dfrac{\partial c}{\partial t} + \mathbf{u}\cdot\nabla c = 0, \quad x \in \mathbb{R}^d, \ t > 0, \\ \\ \qquad\qquad c(X,0) = c^0(x), \end{cases} \tag{2.1}$$

where $c : \mathbb{R}^d \times [0,T] \to \mathbb{R}$, $\mathbf{u} : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$ is a vector-valued function and $c^0(x)$ has compact support defined in a domain $D_0 \subset\subset \mathbb{R}^d$. Next, we introduce the characteristics curves of the first order differential operator

$$\frac{\partial}{\partial t} + u\cdot\nabla,$$

which are the solution (if it exists) of the system of ordinary differential equations

$$\begin{cases} \dfrac{dX(x,s;t)}{dt} = \mathbf{u}(X(x,s;t),t), \\ \\ \qquad X(x,s;s) = x. \end{cases} \tag{2.2}$$

The notation $X(x,s;t)$ of the solution of (2.2) is used in order to make explicit its dependence on the initial condition (x,s). Let $c(X(x,s;t),t)$ denote the value of the variable c at time t on the curves $X(x,s;t)$, the variation of c along such curves is then given as

$$\frac{Dc(X(x,s;t),t)}{Dt} = \frac{\partial c(X(x,s;t),t)}{\partial t} + \frac{dX(x,s;t)}{dt}\cdot\nabla c(X(x,s;t),t) =$$

$$\frac{\partial c(X(x,s;t),t)}{\partial t} + \mathbf{u}(X(x,s;t),t)\cdot\nabla c(X(x,s;t),t).$$

Hence, the Cauchy problem (2.1) can be written as an ordinary differential equation along the characteristics curves, $X(x,s;t)$, of the form

$$\begin{cases} \dfrac{Dc}{Dt} = 0\ , \ \ X(x,s;t) \in \mathbb{R}^d,\ t > 0, \\ \\ \qquad c(X(x,0;0),0) = c^0(x) \end{cases} \tag{2.3}$$

Assuming that $\mathbf{u} \in C([0,T], W^{1,\infty}(\mathbb{R}^d)^d)$ and $c^0(x)$ is sufficiently smooth, problem (2.2) has a unique solution and the solution of (2.3) is then given as

$$c(X(\cdot,t;t+\tau),t+\tau) = c(\cdot,t). \tag{2.4}$$

3 Finite element formulation of the conventional LG method

In the framework of finite elements, two approaches have been proposed to generate a time marching algorithm to approximate the solution $c(x,t)$. One of them is the so called LG method introduced by Douglas and Russell (1982) and Pironneau (1982) and the second one is the weak LG method proposed by Benqué et al. (1982). The realization of these approaches requires the definition of a partition D_h on a domain $D \subset \mathbb{R}^d$ sufficiently large such that $D_0 \subset D$, if necessary D may vary with time t, and then the construction of a finite element space V_h associated to D_h. Specifically, we consider that the closed region $\overline{D} := D \cup \partial D$ is tessellated to produce a quasi-uniform regular partition D_h composed by simplexes with the property that if ∂D is a curved boundary the elements adjacent to the boundary will have at least one curved face (see [11] for the theory of curved elements), whereas the elements in the interior of D will have plane faces. Given the integer $NE > 1$,

$$D_h := \left\{ T_j \subset \overline{D} : D \cup \partial D = \bigcup_{j=1}^{Ne} T_j \right\},$$

and there exist real constants $\sigma > 0$ and $\nu > 0$, such for all j the following quasi-uniformity and regularity conditions, respectively, hold

$$\frac{h}{h_j} < \nu \text{ and } \frac{h_j}{\rho_j} < \sigma,$$

where $h_j :=$diam(T_j), $\rho_j :=$ sup{diam(S), S is a ball contained in T_j} and $h :=$max(T_j) is the mesh size parameter in D_h. We associate with D_h the conforming finite element spaces

$$W_h := \left\{ v_h \in C^0(\overline{D}) : v_h \mid_{T_j} \in P(T_j), \quad \forall T_j \in D_h \right\},$$

$$V_h = H_0^1(D) \cap W_h,$$

with

$$P(T_j) = \left\{ p(x) : \text{for } x \in T_j,\, p(x) = \widehat{p} \circ F_j^{-1}(x),\, \widehat{p} \in P_m(\widehat{T}), \right\}$$

where $P_m(\widehat{T})$ denotes the set of polynomials of degree $\leq m$ defined on the simplex of reference $\widehat{T}$, and $F_j : \widehat{T} \to T_j$ is a mapping of class $C^{m,1}$ when T_j is a curved element of class m, otherwise it is an invertible affine mapping of the form

$$F_j(\widehat{x}) = \mathbf{B}_j \widehat{x} + \mathbf{b}_j, \quad \mathbf{B}_j \in \mathcal{L}(\mathbb{R}^d) \text{ and } \mathbf{b}_j \in \mathbb{R}^d. \tag{3.1}$$

Also, we define a partition of the interval $[0,T]$ into subintervals $I_n := [t_n, t_{n+1}]$ of length $\Delta t = t_{n+1} - t_n$ for all n, $0 = t_0 < t_1 < ... < t_N = T$. The approximate solution $c_h^n(x) \in V_h$ at time t_n is given as

$$c_h^n(x) = \sum_{i=1}^{M} C_i^n \phi_i(x),$$

where $C_i^n := c_h^n(x_i)$, x_i being the i-th mesh-point in D_h, M denotes the number of mesh-points of the partition D_h, and $\{\phi_i\}_{i=1}^M$ is the set of global basis functions of V_h. Denoting by $X_h^{n,n+1}(x) : = X_h(x, t_{n+1}; t_n)$ an approximation to the foot at time t_n (also known as departure point) of the characteristic curve $X(x, t_{n+1}; t)$, the conventional LG method calculates the approximation $c_h^{n+1}(x) \in V_h$ as

$$\int_D c_h^{n+1}(x)\phi_i(x)dx = \int_D c_h^n(X_h^{n,n+1}(x))\phi_i(x)dx, \tag{3.2}$$

whereas the weak LG method does

$$\int_D c_h^{n+1}(x)\phi_j(x)dx = \int_D c_h^n(X_h^{n,n+1}(x))\phi_i(x)dX_h^{n,n+1}(x). \tag{3.3}$$

In matrix form the approximation $c_h^{n+1}(x)$ is calculated as

$$\begin{cases} \mathbf{M}[\mathbf{C}^{n+1}] = \mathbf{R} \quad \text{(conventional LG)}, \\ \\ \mathbf{M}[\mathbf{C}^{n+1}] = \widetilde{\mathbf{R}} \quad \text{(weak LG)}, \end{cases} \tag{3.4}$$

where $\mathbf{M}$ is the so called mass matrix, the elements m_{ij} of which are given by $m_{ij} = \int_D \phi_i(x)\phi_j(x)dx,\ i, j = 1, 2, ..., M$, and the M-dimensional vectors $\mathbf{R} := (r_1, ..., r_M)^T$ and $\widetilde{\mathbf{R}} := (\widetilde{r}_1, ..., \widetilde{r}_M)^T$, with r_i and $\widetilde{r}_i$ being the right hand sides of (3.2) and (3.3) for the basis function $\phi_i(X)$, respectively. The most crucial point in both approaches is the evaluation of the integrals on the right hand sides, and consequently, the calculation of the departure points. Hereafter, we shall only consider the LG method.

3.1 Calculation of the integrals in the conventional LG method

The evaluation of $\int_D c_h^n(X_h^{n,n+1}(x))\phi_i(x)dx$ is usually done numerically by applying a quadrature rule of high order to maintain both the stability and the accuracy of the method when the integrals are calculated exactly, see Morton et al. (1988). Noting that we can write

$$\int_D c_h^n(X_h^{n,n+1}(x))\phi_i(x)dx = \sum_{j=1}^{NE} \int_{T_j} c_h^n(X_h^{n,n+1}(x))\phi_i(x)dx \quad \text{for all} \quad j$$

and considering that $c_h^n \in V_h$ is expressed as

$$c_h^n(x) = \sum_{k=1}^{M} C_k^n \phi_k(x),$$

we can write the restriction of $c_h^n(x)$ to the element T_j as

$$c_h^n(x) \mid_{T_j} = \sum_{l=1}^{ne} C_{l(j)}^n \varphi_l^{(j)}(x),$$

where $l(j)$ denotes the (global) number of the node of the mesh D_h that is the l-th node of the element T_j, $\{\varphi_l^{(j)}\}_{l=1}^{ne}$ is the set of the local basis functions for the element T_j and ne denotes the number of nodes defining T_j. Then for each T_j the integrals

$$\int_{T_j} c_h^n(X_h^{n,n+1}(x))\phi_i(x)dx$$

are evaluated by the formula

$$\sum_{l=1}^{ne} C_{l(j)}^n \int_{T_j} \varphi_l^{(i)}(X_h^{n,n+1}(x))\varphi_p^{(j)}(x)dx \quad (1 \leq p \leq ne),$$

here, $\varphi_l^{(i)}$ is the l-th local basis function of the element T_i where the point $X_h^{n,n+1}(x)$ is located. As it is customary in finite element practice, we make use of the reference element $\widehat{T}$, through the bijective transformation $F_j : \widehat{T} \to T_j$, to calculate such integrals; thus, we set

$$\int_{T_j} \varphi_l^{(i)}(X_h^{n,n+1}(x))\varphi_p^{(j)}(x)dx = \int_{\widehat{T}} \widehat{\varphi}_l^{(i)}(\widehat{x})\widehat{\varphi}_p(\widehat{x}) \left| \frac{\partial F_k}{\partial \widehat{x}} \right| d\widehat{x}, \tag{3.5}$$

where $\widehat{\varphi}_l^{(i)}(\widehat{x}) := \varphi_l^{(i)}(X_h^{n,n+1} \circ F_j(\widehat{x}))$, $\{\widehat{\varphi}_p\}_{p=1}^{ne}$ denotes the set of basis function in $\widehat{T}$ and $\left| \frac{\partial F_k}{\partial \widehat{x}} \right| > 0$ is the determinant of the Jacobian of F_j. We explain what is $\widehat{\varphi}_l^{(i)}(\widehat{x})$. For this purpose, we note that there exists one and only one $z := X_h^{n,n+1} \circ F_j(\widehat{x}) \in T_i$ and one and only one $F_i^{-1} : T_i \to \widehat{T}$ such that $\varphi_l^{(i)}(X_h^{n,n+1}(x)) = \varphi_l^{(i)}(X_h^{n,n+1} \circ F_j(\widehat{x})) = \widehat{\varphi}_l \circ F_i^{-1}(z)$, then we set $\widehat{\varphi}_l^{(i)}(\widehat{x}) = \widehat{\varphi}_l \circ F_i^{-1}(z)$. We apply a quadrature rule to approximate the integral over $\widehat{T}$ as follows:

$$\int_{\widehat{T}} \widehat{\varphi}_l^{(l)}(\widehat{x})\widehat{\varphi}_p(\widehat{x}) \left| \frac{\partial F_k}{\partial \widehat{x}} \right| d\widehat{x} \simeq meas(T_k) \sum_{g=1}^{nqp} \varpi_g \widehat{\varphi}_l^{(l)}(\widehat{x}_g)\widehat{\varphi}_p(\widehat{x}_g),$$

where $\{\varpi_g\}$ and $\{\widehat{x}_g\}$ denote the sets of weights and points, respectively, of the quadrature formula. Note that for $x_g \in T_j = F_j(\widehat{x}_g)$ and $\widehat{z}_g \in \widehat{T} = F_i^{-1} \circ X^{n,n+1}(x_g)$,

$$c_h^n(X_h^{n,n+1}(x_g)) = \sum_{l=1}^{ne} C_{l(i)}^n \varphi_l^{(i)}(X_h^{n,n+1}(x_g)) = \sum_{l=1}^{ne} C_{l(i)}^n \widehat{\varphi}_l^{(i)}(\widehat{x}_g) = \sum_{l=1}^{ne} C_{l(i)}^n \widehat{\varphi}_l(\widehat{z}_g).$$

Hence, an algorithmic presentation of the numerical procedure to compute the integrals at any time instant t_n and to calculate c_h^{n+1} is the following.

Algorithm 1 (Conventional LG algorithm)

Assume that u^n and c_h^n are known, and let $\widehat{x}_g$ and ϖ_g be the quadrature points and their associated weights of the reference element $\widehat{T}$. Then:

(1) Calculate the mass matrix M.
(2) *For $j = 1, 2 \ldots NE$*
For $g = 1, 2 \ldots nqp$
(2.1) Calculate

$$x_g = F_j(\widehat{x}_g).$$

(2.2) Calculate $X_h^{n,n+1}(x_g)$ by solving (2.2) with initial conditions x_g.
(2.3) Find the element T_i containing the point $X_h^{n,n+1}(x_g)$ and calculate

$$\widehat{z}_g = F_i^{-1} \circ X_h^{n,n+1}(x_g)\,.$$

(2.4) Calculate

$$c_h^n(X_h^{n,n+1}(x_g)) = \sum_{l=1}^{ne} C_{l(i)}^n \widehat{\varphi}_l(\widehat{z}_g).$$

(2.5) For $p = 1, \ldots ne$
(a) Calculate

$$meas(T_j) \sum_{g=1}^{ngp} \varpi_g c_h^n(X_h^{n,n+1}(x_g))\widehat{\varphi}_p(\widehat{x}_g).$$

(b) Assemble these values into the right hand side vector $R^n := (r_1^n, \ldots, r_M^n)^T$
(3) Calculate $[\mathbf{C}^{n+1}]$ by solving (3.2).
(4) Define $c_h^{n+1} \in V_h$ as

$$c_h^{n+1}(x) = \sum_{i=1}^{M} C_i^{n+1} \phi_i(x).$$

Some remarks are now in order.

Remark 1. *It is worth noting that in the evaluation of the integrals it is necessary to solve nqp times the system (2.2) followed by the corresponding searching of the elements containing the points $X_h^{n,n+1}(x_g)$ and the calculation of $c_h^n(X_h^{n,n+1}(x_g))$. As we will see below, the theoretical analysis, supported by numerical experiments, shows that the error of the LG method depends on the accuracy of the numerical solution of the system (2.2), then it is wise to employ a numerical method of order ≥ 2 to calculate the points $X_h^{n,n+1}(x_g)$.*

Remark 2. *The number of quadrature points, nqp, may be quite large, in particular, in three dimensional problems, because the use of high order rules is recommended to maintain the stability and convergence properties that the method possesses when the integrals are calculated exactly.*

Remark 3. *Steps (2.2), (2.3) and (2.4) of this procedure may be time consuming if they are not properly implemented, in particular, when the partition D_h is unstructured as is usual in finite element calculations.*

3.2 Calculation of the departure points $X_h^{n,n+1}(x_g)$ and interpolation of the solution at them

We focus on the realization of steps (2.2)-(2.4) of the LG algorithm. Several methods have been proposed to calculate an approximate solution to (2.2), such as explicit Runge-Kutta methods of order ≥2 or a fixed point implicit multi-step method of order 2. Here we shall describe a Runge-Kutta method of order 3 and the fixed point implicit multi-step method.

The Runge-Kutta method of order 3

The algorithmic presentation of this method for the calculation of the departure points $X_h^{n,n+1}(x_g)$ is as follows.

Algorithm 2

Assuming that at time t_{n+1} we know $\mathbf{u}^{n+1}$, then for each integration point x_g, $1 \leq g \leq nqp$, calculate

$$\begin{cases} K_1 = \mathbf{u}(x_g, t_{n+1}), \\ K_2 = \mathbf{u}(x_g - \Delta t K_1, t_{n+1} - \Delta t), \\ K_3 = \mathbf{u}\left(x_g - \dfrac{\Delta t K_1}{4} - \dfrac{\Delta t K_2}{4}, t_{n+1} - \dfrac{\Delta t}{2}\right), \\ X_h^{n,n+1}(x_g) = x_g - \Delta t\left(\dfrac{K_1}{6} + \dfrac{K_2}{6} + \dfrac{4K_3}{6}\right). \end{cases} \tag{3.6}$$

In this algorithm the crucial steps are the calculations of K_1, K_2 and K_3 that have to be done with care if we want to get an accurate result. For this purpose, we must notice that in general the velocity $\mathbf{u}$ is not an analytical function, but a numerical solution provided by an external source at the mesh points $\{x_i\}$ at time steps t_n, n integer. Therefore, $\mathbf{u}(\cdot, t_{n+1} - \frac{\Delta t}{2})$ is an unknown as well as $\mathbf{u}(x_g, t_{n+1})$ and $\mathbf{u}(x_g - \Delta t K_1, t_{n+1} - \Delta t)$ because in general the points x_g and $x_g - \Delta t K_1$ do not coincide with a mesh point. Thus, $\mathbf{u}(\cdot, t_{n+1} - \frac{\Delta t}{2})$ is approximated by the interpolation formula

$$\mathbf{u}(\cdot, t_{n+1} - \frac{\Delta t}{2}) = \frac{6\mathbf{u}(\cdot, t_{n+1}) + 6\mathbf{u}(\cdot, t_n) - \mathbf{u}(\cdot, t_{n-1})}{8}$$

and $\mathbf{u}(x_g, t_{n+1})$, $\mathbf{u}(x_g - \Delta t K_1, t_{n+1} - \Delta t)$ and $\mathbf{u}\left(x_g - \frac{\Delta t K_1}{4} - \frac{\Delta t K_2}{4}, t_{n+1} - \frac{\Delta t}{2}\right)$ are calculated by finite element interpolation from the values of $\mathbf{u}(\cdot, t_{n+1})$, $\mathbf{u}(\cdot, t_{n+1} - \Delta t)$ and $\mathbf{u}\left(\cdot, t_{n+1} - \frac{\Delta t}{2}\right)$, respectively, at the vertices of the elements where the points x_g, $x_g - \Delta t K_1$ and $x_g - \frac{\Delta t K_1}{4} - \frac{\Delta t K_2}{4}$ are located. In structured quadrilateral meshes it is very easy to search for the elements containing those points; however, in unstructured and/or simplicial meshes this task is more complicated and should be done with much care, otherwise the method may become inefficient. In Allievi and Bermejo (1997) a search-locate algorithm to identify the elements containing the points x_g, $x_g - \Delta t K_1$ and $x_g - \frac{\Delta t K_1}{4} - \frac{\Delta t K_2}{4}$ is described, and the finite element interpolations at them is performed. Such an algorithm uses the Newton method to invert the bijective map from the reference element into a given mesh element together with a criterium to move from element to element in the mesh.

Remark 4. *When $n = 0$, the calculation of $X_h^{n,n+1}(x_g)$ by (3.6) is carried out by the second order Runge-Kutta formula*

$$X_h^{0,1}(x_g) = x_g - \Delta t \left(\frac{K_1}{2} + \frac{K_2}{2} \right)$$

because the velocity is not defined at time t_{-1}.

Remark 5. *For all x_g, the departure points $X_h^{n,n+1}(x_g)$ cannot leave the computational domain through the solid boundaries because on such boundaries either $\mathbf{u} = 0$ or $\mathbf{u} \cdot \mathbf{n} = \mathbf{0}$, $\mathbf{n}$ being the outwards normal at the boundary points, so that it can be proven that the trajectories of (2.2) cannot cross the solid boundaries. However, in many cases, in particular when Δt is not small enough, the numerical errors cause that the points $x_g - \Delta t K_1$ and $x_g - \frac{\Delta t K_1}{4} - \frac{\Delta t K_2}{4}$ to be outside the computational domain when the points x_g are in elements close to the solid boundaries of D. One way to correct this is to apply (3.6) for such points x_g halving the time step until getting that $x_g - \Delta t K_1$, $x_g - \frac{\Delta t K_1}{4} - \frac{\Delta t K_2}{4}$ and $X_h^{n,n+1}(x_g)$ remain inside the computational domain; in doing so, the extrapolation formulas to calculate the velocity $\mathbf{u}(\cdot, t_n + \frac{\Delta t}{2^k})$, $k = 0, 1, 2, \ldots k_{\max}$, have to be corrected accordingly. The search-locate algorithm of Allievi and Bermejo (1997) ascertains when a point leaves the domain.*

Remark 6. *If the velocity $\mathbf{u}$ is not known at time t_{n+1}, as it happens, for instance, when LG methods are used to solve the time dependent incompressible Navier-Stokes equations, then $\mathbf{u}(\cdot, \mathbf{t}_{n+1})$ is extrapolated by the third order formula*

$$\mathbf{u}(\cdot, t_{n+1}) = 3\mathbf{u}(\cdot, t_n) - 3\mathbf{u}(\cdot, t_{n-1}) + \mathbf{u}(\cdot, t_{n-2}),$$

thus requiring to store the velocity values at time instants t_n, t_{n-1} and t_{n-2}. This fact plus the accuracy problems sometimes involved in extrapolation formulas are the reasons why some authors prefer to use a second order multi-step method, instead of the Runge-Kutta methods of order higher than 2, to calculate the departure points when solving the Navier-Stokes equations with first or second order in time LG methods.

A fixed point implicit multi-step method of order 2

Since the solution of (2.2) for $x = x_g$ can also be expressed by the formula

$$X(x_g, t_{n+1}; t_n) = x_g - \int_{t_n}^{t_{n+1}} \mathbf{u}(X(x_g, t_{n+1}; t), t)dt,$$

then setting

$$\alpha_g = x_g - X(x_g, t_{n+1}; t_n)$$

and approximating the integral by the mid-point rule, we obtain a formula to approximate α_g up to order $O(\Delta t^3)$ such as

$$\alpha_g = \Delta t \mathbf{u}(X(x_g, t_{n+1}; t_n + \frac{\Delta t}{2}), t_n + \frac{\Delta t}{2}).$$

Moreover, using the second order approximations

$$X(x_g, t_{n+1}; t_n + \frac{\Delta t}{2}) = \frac{1}{2}(x_g + X(x_g, t_{n+1}; t_n) = x_g - \frac{1}{2}\alpha_g$$

and

$$\mathbf{u}(\cdot, t_n + \frac{\Delta t}{2}) = \frac{3}{2}\mathbf{u}(\cdot, t_n) - \frac{1}{2}\mathbf{u}(\cdot, t_{n-1}),$$

we have that

$$\alpha_g = \Delta t \left(\frac{3}{2}\mathbf{u}(x_g - \frac{1}{2}\alpha_g, t_n) - \frac{1}{2}\mathbf{u}(x_g - \frac{1}{2}\alpha_g, t_{n-1}) \right).$$

This is an implicit equation the solution of which is calculated via the following fixed point iterative procedure.

Algorithm 3

Given Δt , *the integer number* $k_{\max}$ *and the real number* ε, *and assuming that* $\mathbf{u}^n$ *and* $\mathbf{u}^{n-1}$ *are known, then for each integration point* x_g, $1 \leq g \leq nqp$, *calculate*

(1)

$$\alpha_g^{(0)} = \Delta t \left(\frac{3}{2}\mathbf{u}(x_g, t_n) - \frac{1}{2}\mathbf{u}(x_g, t_{n-1}) \right).$$

(2) For $k = 0, 1, \ldots,$

$$\alpha_g^{(k+1)} = \Delta t \left(\frac{3}{2}\mathbf{u}(x_g - \frac{1}{2}\alpha_g^{(k)}, t_n) - \frac{1}{2}\mathbf{u}(x_g - \frac{1}{2}\alpha_g^{(k)}, t_{n-1}) \right).$$

The iterative procedure stops when $k = k_{\max}$ *or when the stopping criterium*

$$\frac{\left| \alpha_g^{(k+1)} - \alpha_g^{(k)} \right|}{\left| \alpha_g^{(k)} \right|} \leq \varepsilon$$

is satisfied.

(3) Set

$$X_h^{n,n+1}(x_g) = x_g - \alpha_g^{(k+1)}.$$

Remark 7. *It is easy to show that the above procedure converges if*

$$\Delta t \max_{(x,t_n)\in B_j\times(0,T)} |\nabla \mathbf{u}^n(x)| < 2,$$

where B_j is a neighborhood of the point x_g such that $X_h^{n,n+1}(x_g) \in B_j$.

Remark 8. *At each iteration we have to identify the element that contains $x_g - \frac{1}{2}\alpha_g^{(k)}$. As in the Runge-Kutta method, this is done by the search-locate algorithm of Allievi and Bermejo (1997).*

Remark 9. *If Δt is so large that either the iterative procedure does not converge or the points leave the computational domain, then we successively halve m times Δt, thus yielding the following adaptive iterative procedure.*

Algorithm 4

Given Δt, the integer number $k_{\max}$ and the real number ε, assuming that $\mathbf{u}^n$ is known and setting m=1, then for each integration point x_g, $1 \leq g \leq nqp$:

(1) Calculate

$$\alpha_g^{(0)} = 2^{-m+1}\Delta t\mathbf{u}(x_g, t_n).$$

If $x_g - \dfrac{1}{2}\alpha_g^{(0)}$ leaves the computational domain through a solid boundary, set $m = m+1$ and repeat (1).

(2) For $k = 0, 1, \ldots, k_{\max}$ calculate

$$\alpha^{(k+1)} = 2^{-m+1}\Delta t\mathbf{u}(x_g - \frac{1}{2}\alpha_g^{(k)}, t_{n+1} - \frac{\Delta t}{2^m}).$$

(a) If $x_g - \dfrac{1}{2}\alpha_g^{(0)}$ leaves the computational domain through a solid boundary, set $m = m+1$ and repeat (1).

(b) If $k = k_{mzx}$ and

$$\frac{\left|\alpha_g^{(k+1)} - \alpha_g^{(k)}\right|}{\left|\alpha_g^{(k)}\right|} > \varepsilon,$$

then *set $m = m+1$ and repeat (1).*

(c) If

$$\frac{\left|\alpha_g^{(k+1)} - \alpha_g^{(k)}\right|}{\left|\alpha_g^{(k)}\right|} \leq \varepsilon$$

stop the iterations.
(3) Set

$$X_h(x_g, t_{n+1}; t_{n+1} - \frac{2\Delta t}{2^m}) = x_g - \alpha_g^{(k+1)} \left(t_{n+1} - \frac{\Delta t}{2^m} \right).$$

(3) For $l = m-1, .., 1$ *calculate*

$$X_h \left(x_g, t_{n+1}; t_{n+1} - \frac{2\Delta t}{2^l} \right)$$
$$= x_g - 2^{-m+1}\Delta t \mathbf{u} \left(X_h \left(x_g, t_{n+1}; t_{n+1} - \frac{\Delta t}{2^l} \right), t_{n+1} - \frac{\Delta t}{2^l} \right).$$

Note that when $l = 1$, $X_h^{n,n+1}(x_g) = X_h \left(x_g, t_{n+1}; t_{n+1} - \frac{2\Delta t}{2^l} \right)$. In the above formulas

$$\mathbf{u}(\cdot, t_{n+1} - \frac{\Delta t}{2^k}) = (2 - 2^{-k})\mathbf{u}(\cdot, t_n) - (1 - 2^{-k})\mathbf{u}(\cdot, t_{n-1}),\ 1 \le k \le m.$$

It can be shown that this iterative procedure converges if

$$\Delta t \max_{(x,t_n) \in B_j \times (0,T)} |\nabla \mathbf{u}^n(x)| < 2^m.$$

4 A modified LG method

As we said in the previous section, the conventional LG method (also the weak LG method) requires high order quadrature rules and, consequently, the calculation of many departure points per element. The latter calculation is time consuming, in particular in 3D problems with simplicial meshes. To alleviate this issue, we have recently proposed [8] and [7] a modification of the conventional LG method that, while maintaining its order of convergence when P_1 and P_2 elements are used, reduces significantly the CPU time of the method, because the number of departure points to be calculated in every time step is equal to the number of mesh nodes that are vertices of the mesh elements. The idea of the method, which is graphically represented in Figure 1, can be explained as follows. At any time t_n we can construct an element $T_j^{n,n+1} \subset \overline{D}$ associated with the element T_j of the partition D_h as follows: $T_j^{n,n+1} := \{y = X_h^{n,n+1}(x),\ x \in T_j\}$, so that $T_j^{n,n+1} = X_h^{n,n+1}(T_j)$; moreover, it can be shown, $\overline{D} = \bigcup_j T_j^{n,n+1}$. Next, we can also define a quasi-isometric map of class $C^{k-1,1}$, $F_j^{n,n+1} : \widehat{T} \to T_j^{n,n+1}$ such that for all $\widehat{x} \in \widehat{T}$ there is one and only one

$$y = F_j^{n,n+1}(\widehat{x}) = X_h^{n,n+1} \circ F_j(\widehat{x}) = X_h^{n,n+1}(x). \tag{4.1}$$

Note that due to the properties of the maps $F_j^{n,n+1}$ and F_j there exists $(F_j^{n,n+1})^{-1}$. In relation with $T_j^{n,n+1}$ we consider the simplex $\widetilde{T}_j^{n,n+1}$, the vertices of which are the points $\{X_h^{n,n+1}(a_{j1}), \ldots, X_h^{n,n+1}(a_{jd+1})\}$, the latter are images of the vertices of T_j, $\{a_{j1}, \ldots, a_{jd+1}\}$,

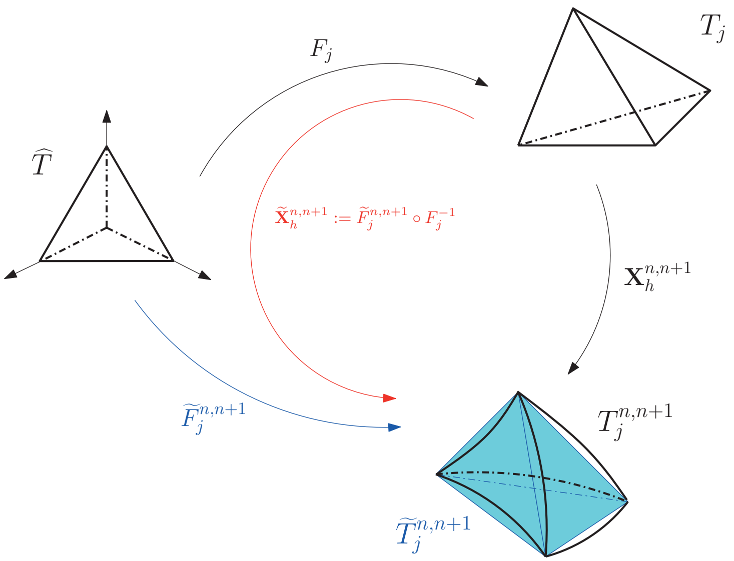

Figure 1: The mappings for the formulation of the MLG method

and then define the invertible affine map, $\widetilde{F}_j^{n,n+1} : \widehat{T} \to \widetilde{T}_j^{n,n+1}$ as follows: for all $\widehat{x} \in \widehat{T}$ there is one and only one

$$\widetilde{y} = \widetilde{F}_j^{n,n+1}(\widehat{x}) = \widetilde{\mathbf{B}}_j^{n,n+1}\widehat{x} + \widetilde{\mathbf{b}}^{n,n+1},\ \widetilde{\mathbf{B}}_j^{n,n+1} \in \mathcal{L}(\mathbb{R}^d) \text{ and } \widetilde{\mathbf{b}}_j^{n,n+1} \in \mathbb{R}^d. \tag{4.2a}$$

Note that $\widetilde{T}_j^{n,n+1}$ is a linear approximation to $T_j^{*n,n+1}$ and hence

$$\widetilde{F}_j^{n,n+1} = \widehat{I}F_j^{n,n+1}, \tag{4.2b}$$

where $\widehat{I}$ denotes the linear interpolant on $\widehat{T}$. At this point, it is important to remark that at any time t_n, for each $\widehat{x} \in \widehat{T}$, and consequently for each $x \in T_j$, with $x = F_j(\widehat{x})$, we can associate two points, namely, the point $y = F_j^{n,n+1}(\widehat{x}) = X_h^{n,n+1}(x) \in T_j^{n,n+1}$ and the point $\widetilde{y} = \widetilde{F}_j^{n,n+1}(\widehat{x}) = \widetilde{F}_j^{n,n+1} \circ F_j^{-1}(x) \in \widetilde{T}_j^{n,n+1}$; hereafter we shall use $\widetilde{X}_h^{n,n+1}(x) := \widetilde{F}_j^{n,n+1} \circ F_j^{-1}(x)$ to denote $\widetilde{y}$. The modified LG method consists of calculating $c_n^{n+1} \in V_h$ as solution of

$$\int_D c_h^{n+1}(x)\phi_i(x)dx = \int_D c_h^n(\widetilde{X}_h^{n,n+1}(x))\phi_i(x)dx. \tag{4.3}$$

Remark 10. *For simplicity in the presentation of the developments that follow, we assume that the boundary ∂D is Lipschitz continuous and formed by plane faces (straight sides when $d = 2$). In case we allow curved boundary, there will be boundary elements $T_j^{n,n+1}$ having* *a curved face and, therefore, $\widetilde{T}_j^{n,n+1} \cap \partial D \neq T_j^{n,n+1} \cap \partial D$; in such a case, we should extend the functions defined on D by zero outside $\overline{D}$ if $\widetilde{T}_j^{n,n+1} \cap \overline{D}$ is not included in $\overline{D}$.*

Looking at (4.3) we shall have to calculate integrals of the form

$$\int_{T_j} \varphi_l^{(i)}(\widetilde{X}_h^{n,n+1}(x))\varphi_p^{(j)}(x)dx = \int_{\widehat{T}} \widehat{\varphi}_l(\widehat{\widetilde{z}})\widehat{\varphi}_p(\widehat{x}) \left|\frac{\partial F_j}{\partial \widehat{x}}\right| d\widehat{x},$$

where $\widehat{\widetilde{z}} := F_i^{-1} \circ \widetilde{F}_j^{n.n+1}(\widehat{x})$, T_i being the element containing $\widetilde{F}_j^{n.n+1}(\widehat{x})$. Similarly as we did in the standard LG method, the integrals over the element $\widehat{T}$ are then approximated by a high order quadrature rule as

$$\int_{\widehat{T}} \widehat{\varphi}_l(\widehat{\widetilde{z}})\widehat{\varphi}_p(\widehat{x}) \left|\frac{\partial F_j}{\partial \widehat{x}}\right| d\widehat{x} \simeq meas(T_j)\sum_{k=1}^{nqp} \varpi_k \widehat{\varphi}_l(\widehat{\widetilde{z}}_k)\widehat{\varphi}_p(\widehat{x}_k),$$

where $\widehat{\widetilde{z}}_k = F_i^{-1} \circ \widetilde{F}_j^{n.n+1}(\widehat{x})$, and noting that we have to calculate now

$$c_h^n(\widetilde{X}_h^{n,n+1}(x_g)) = \sum_{l=1}^{ne} C_{l(i)}^n \varphi_l^{(i)}(\widetilde{X}_h^{n,n+1}(x_g)) = \sum_{l=1}^{ne} C_{l(i)}^n \widehat{\varphi}_l(\widehat{\widetilde{z}}_k)$$

instead of $c_h^n(X^{n,n+1}(x_g))$. The procedure to approximate the integrals at time t_n in the modified LG methods is as follows:

Algorithm 5 (Modified LG algorithm)

Assume that $\mathbf{u}^n$ and c_h^n are known, and let $\widehat{x}_g$ and ϖ_g be the quadrature points and their associated weights of the reference element $\widehat{T}$. Then:

(1) Calculate the mass matrix M.

(2) For $j = 1, 2 \ldots NE$

(2..1) Calculate the element $\widetilde{T}_j^{n,n+1}$ by solving (2.2) taking as initial conditions the vertices $\{a_{ij}\}_{i=1}^{d+1}$ of the element T_j.

(2.2) For $k = 1, 2 \ldots nqp$

(a) Calculate

$$\widetilde{X}^{n,n+1}(x_k) = \widetilde{B}_j^{n,n+1}\widehat{x}_k + \widetilde{b}_j^{n,n+1}.$$

(b) Find the element T_i containing the point $\widetilde{X}^{n,n+1}(x_k)$ and calculate

$$\widehat{\widetilde{z}}_k = F_i^{-1}(\widetilde{X}^{n,n+1}(x_k)).$$

(c) Calculate

$$c_h^n(\widetilde{X}^{n,n+1}(x_k)) = \sum_{l=1}^{ne} C_{l(i)}^n \widehat{\varphi}_l(\widehat{\widetilde{z}}_k).$$

(3) For $p = 1, \ldots ne$

(3.1) Calculate

$$meas(T_j)\sum_{k=1}^{ngp}\varpi_k c_h^n(\widetilde{X}^{n,n+1}(x_k))\widehat{\varphi}_p(\widehat{x}_k).$$

(3.2) Assemble these values into the right hand side column vector.

(4) Calculate $[\mathbf{C}^{n+1}]$ *by solving (*3.4*).*

(5) Define $c_h^{n+1}\in V_h$ *as*

$$c_h^{n+1}(x)=\sum_{i=1}^{M}C_i^{n+1}\phi_i(x).$$

5 Analysis of LG methods for linear convection equations

In this section we shall study the convergence of the LG methods. First, under the assumption that the integrals are calculated exactly, we establish the L^2-norm stability of the LG methods. To make clear the steps of our analysis we assume that the departure points are calculated exactly. Some preliminary results concerning the solution $X(x,t_{n+1};t)$ of (2.2), which are well known in the theory of ODE systems, are necessary in our developments.

Lemma 1. *Assume that* $u\in L^\infty\ (0,T;W^{k,\infty}(D)^d),\ k\geq 1$*. Then for any* n, $0\leq n\leq N-1$*, there exists a unique solution* $t\to X(x,t_{n+1};t)$ $(t\in[t_{n-1},t_{n+1}[\subset[0,T])$ *of (*2.2*) such that* $X(x,t_{n+1};t)\in W^{1,\infty}(0,T;W^{k,\infty}(D)^d)$*. Furthermore, let the multi-index* $\alpha\in N^d$*, then for all* α*, such that* $1\leq\mid\alpha\mid\leq k$, $\partial_x^{\boldsymbol{\alpha}}X_i(x,t_{n+1},t)\in C^0([0,T];L^\infty(D\times[0,T]))$*,* $1\leq i\leq d$*.*

Lemma 2. *Suppose the assumptions of Lemma* 1 *hold true and* $u\mid_{\partial D}=0$*. For* $|s-t|$ *sufficiently small,* $x\to X(x,s;t)$ *defines a quasi-isometric map of class* $C^{k-1,1}$ *of* $\overline{D}$ *onto* $\overline{D}$ *with Jacobian determinant* $J(x,s;t)\in C^0([0,T];L^\infty(D\times[0,T]))$ *satisfying*

$$\exp(-C_u\,|s-t|)\leq J(x,s;t)\leq\exp(C_u\,|s-t|),\tag{5.1}$$

where $C_u=\|div\mathbf{u}\|_{L^\infty(D\times(0,T))}$

Moreover,

$$K_u^{-1}\mid x-y\mid\leq\mid X(x,s;t)-X(y,s;t)\mid\leq K_u\mid x-y\mid,\tag{5.2}$$

where $K_u=\exp(\mid s-t\mid\mid\nabla u\mid_{L^\infty(0;T,(W^{1,\infty}(D))^d)})$*.* $\mid a-b\mid$ *denotes the Euclidean distance between the points* $a,\ b\in\mathbb{R}$. For a proof of (2) see Süli (1988).

We make the following hypothesis for the parameters Δt and h.

(H1) Given the parameters h_0 and $\Delta t_0<1$ sufficiently small, $0<h<h_0$ and $0<\Delta t<\Delta t_0$.

5.1 Stability and convergence of the conventional LG method

To prove the stability in the L^2-norm of the solution c_h^{n+1} obtained by the conventional LG method, we assume that the integrals in (3.2) are calculated exactly. This assumption is strong in the sense that is very difficult, and consequently computationally no competitive, to calculate exactly the integral on the right hand side because the integrand is the product of two continuous functions defined in two different meshes, so that, one has to resort to quadrature rules of positive weights to calculate the integrals. While it is easy to calculate exactly the left hand side integral because is the integral of polynomial functions defined on the same mesh, the problem is that $c_h^n(X_h^{n,n+1}(x)) \in W^{1,\infty}(D)$, and hence for a given h the integral on the right hand side can only be calculated exactly when the number of quadrature points tend to infinite, as one can prove by applying Steinhaus-Banach theorem.

Lemma 3. *Under the assumptions of Lemma 1, there exists a constant C independent of Δt and h such that for all t_n,*

$$\left\|c_h^{n+1}\right\|_{L^2(D)} \leq (1 + C\Delta t) \left\|c_h^n\right\|_{L^2(D)}. \tag{5.3}$$

Proof: Multiplying by C_i^{n+1} on both sides of (3.2) and summing for i, we get

$$\int_D \left(c_h^{n+1}(x)\right)^2 dx = \int_D c_h^n(X^{n,n+1}(x))c_h^{n+1}(x)dx.$$

By Cauchy-Schwarz inequality it follows that

$$\int_D \left(c_h^{n+1}(x)\right)^2 dx \leq \int_D \left(c_h^n(X^{n,n+1}(x))\right)^2 dx.$$

Let $y = X^{n,n+1}(x)$, the by virtue of Lemma 2 it follows that for Δt sufficiently small

$$\int_D \left(c_h^n(X^{n,n+1}(x))\right)^2 dx \leq (1 + C_1\Delta t) \int_D (c_h^n(y))^2 dy.$$

Making use of this inequality we obtain (5.3). Note that $C_1 = C_1(C_u)$, where C_u is the constant of (5.1) which depends on $div\mathbf{u}$. □

The remainder of this section is devoted to the analysis of the convergence. To do so, we need to introduce the L^2-projector $P_h : L^2(D) \to V_h$, and assume that for all $v \in H^{m+1}(D)$ there exists a constant K independent of h such that

$$\left\|v - P_h v\right\|_{H^s(D)} \leq K h^{m+1-s} \left\|v\right\|_{H^{m+1}(D)}, \quad s = 0,\ 1, \tag{5.4}$$

where $H^0(D) = L^2(D)$. Furthermore, we recall that for all $v_h \in V_h$, $P_h v_h = v_h$ and $\left\|P_h v\right\|_{L^2(D)} \leq \left\|v\right\|_{L^2(D)}$. We have the following result.

Theorem 4. *Let $c \in C([0,T]; H^{m+1}(D))$ and the assumptions of Lemma 1 hold. Then there exists a constant C independent of Δt and h such that*

$$\max_n \|c(t_n) - c_h^n\|_{L^2(D)} \leq K h^{m+1} \|c\|_{L^\infty(0,T;H^{m+1}(D))} + \\ C\frac{t_n}{\Delta t}\min\left(1, \frac{\Delta t\,\|\mathbf{u}\|_{L^\infty([0,T]\times D)^d}}{h}\right) h^{m+1}\|c\|_{L^\infty(0,T;H^{m+1}(D))}. \tag{5.5}$$

Proof: Using the notation $a^n(x)$, and if confusion does not arise simply a^n, to denote $a(x,t_n)$ for a function a define in $D \times [0,T]$, we have that for all n the global error $e^n(x) = c^n(x) - c_h^n(x)$ can be expressed as

$$e^n = (c^n - P_h c^n) + (P_h c^n - c_h^n) \equiv \rho^n(x) + \theta_h^n(x), \tag{5.6}$$

where $\theta_h^n(x) \in V_h$ and $\rho^n(x)$ satisfies, by virtue of (5.4), the bound

$$\|\rho^n\|_{H^s(D)} \leq K h^{m+1-s}\|c^n\|_{H^{m+1}(D)}. \tag{5.7}$$

To estimate θ_h^n we make use of the property that for all $v_h \in V_h$, $P_h v_h = v_h$, and write

$$\theta_h^n(x) = P_h\left(c^n(x) - c_h^n(x)\right) = P_h\left(c^{n-1}\circ X^{n-1,n}(x) - c_h^{n-1}\circ X^{n-1,n}(x)\right),$$

so that

$$\|\theta_h^n(x)\|_{L^2(D)} \leq \left\|\left(c^{n-1} - c_h^{n-1}\right)\circ X^{n-1,n}(x)\right\|_{L^2(D)},$$

arguing as in the stability proof and using (5.6) we have

$$\|\theta_h^n(x)\|_{L^2(D)} \leq (1 + C_1\Delta t)\left(\left\|\rho^{n-1}(x)\right\|_{L^2(D)} + \left\|\theta_h^{n-1}(x)\right\|_{L^2(D)}\right).$$

Hence, bounding $\|\rho^{n-1}(x)\|_{L^2(D)}$ by (5.4) with $s = 0$ we obtain that

$$\|\theta_h^n(x)\|_{L^2(D)} - \left\|\theta_h^0(x)\right\|_{L^2(D)} \leq \\ (1 + C_1\Delta t)K\frac{t_n h^{m+1}}{\Delta t}\|c\|_{l^\infty(H^{m+1}(D)} + C_1\Delta t\sum_{i=1}^{n-1}\left\|\theta_h^i(x)\right\|_{L^2(D)}.$$

By virtue of Gronwall inequality and assuming that $\theta_h^0(x) = 0$,

$$\|\theta_h^n(x)\|_{L^2(D)} \leq C\frac{t_n h^{m+1}}{\Delta t}\|c\|_{l^\infty(H^{m+1}(D)}, \tag{5.8}$$

where $C = (1 + C_1\Delta t)K\exp(C_1 t_n)$. This estimate breaks down when $\Delta t \to 0$ for a fixed h. To get around this problem we use the facts that for all n, $P_h\rho^n = 0$, and $\theta_h^n(x) = \theta_h^{n-1}(X^{n-1,n}(x))$, so that we can write that

$$\theta_h^n(x) = \theta_h^{n-1}(X^{n-1,n}(x)) + P_h\left(\rho^{n-1}\circ X^{n-1,n}(x) - \rho^{n-1}(x)\right).$$

And from this expression it follows that

$$\|\theta_h^n(x)\|_{L^2(D)} \leq \left\|\theta_h^{n-1}(X^{n-1,n}(x))\right\|_{L^2(D)} + \left\|\rho^{n-1} \circ X^{n-1,n}(x) - \rho^{n-1}(x)\right\|_{L^2(D)}.$$

As above,

$$\left\|\theta_h^{n-1}(X^{n-1,n}(x))\right\|_{L^2(D)} \leq (1 + C_1\Delta t)\left\|\theta_h^{n-1}(x)\right\|_{L^2(D)}.$$

To estimate the second term we note that

$$\rho^{n-1}(x) - \rho^{n-1} \circ X^{n-1,n}(x) = \int_{t_{n-1}}^{t_n} \frac{d\rho(X(x,t_n;t),t_{n-1})}{dt} dt,$$

and by Cauchy-Schwarz inequality we get

$$\left|\rho^{n-1} - \rho^{n-1} \circ X^{n-1,n}\right|^2 \leq \Delta t \int_{t_{n-1}}^{t_n} |\mathbf{u}(X(x,t_n;t),t) \cdot \nabla\rho(X(x,t_n;t),t_{n-1})|^2 \, dt,$$

so that

$$\|\rho^{n-1} - \rho^{n-1} \circ X^{n-1,n}\|_{L^2(D)}^2 \leq \Delta t \int_D \int_{t_{n-1}}^{t_n} |\mathbf{u}(X(x,t_n;t),t) \cdot \nabla\rho(X(x,t_n;t),t_{n-1})|^2 \, dtdx \leq$$

$$\Delta t \|\mathbf{u}\|_{L^\infty(0,T;D)^d}^2 \int_{t_{n-1}}^{t_n} \int_D |\nabla\rho(X(x,t_n;t),t_{n-1})|^2 \, dxdt \leq \text{ (setting } y = X(x,t_n;t))$$

$$\Delta t \|\mathbf{u}\|_{L^\infty((0,T)\times D)^d}^2 \int_{t_{n-1}}^{t_n} \int_D |\nabla\rho(y,t_{n-1})|^2 \left(J^{t,n}\right)^{-1} dydt \leq \text{ (by (5.1))}$$

$$(1 + C_1\Delta t)\Delta t^2 \|\mathbf{u}\|_{L^\infty((0,T)\times D)^d}^2 \|\nabla\rho^{n-1}\|_{L^2(D)}^2,$$

where $J^{t,n}$ denotes the Jacobian determinant of the mapping $x \to X(x,t_n;t)$. Collecting these bounds and using (5.4) with $s = 1$ to bound $\|\nabla\rho^{n-1}\|_{L^2(D)}^2$, it follows that

$$\|\theta_h^n\|_{L^2(D)} - \left\|\theta_h^0\right\|_{L^2(D)} \leq C_1\Delta t \textstyle\sum_{i=1}^{n-1} \left\|\theta_h^i\right\|_{L^2(D)} +$$

$$(1 + C_1\Delta t)K\left(\frac{\Delta t \|\mathbf{u}\|_{L^\infty((0,T)\times D)^d}}{h}\right) h^{m+1} \|c^{n-1}\|_{H^{m+1}(D)}.$$

Applying Gronwall inequality and taking $\theta_h^0 = 0$, we obtain that

$$\|\theta_h^n\|_{L^2(D)} \leq C\frac{t_n}{\Delta t}\left(\frac{\Delta t \|\mathbf{u}\|_{L^\infty((0,T)\times D)^d}}{h}\right) h^{m+1} \|c\|_{L^\infty(0,T;H^{m+1}(D))}. \tag{5.9}$$

From (5.8) and (5.9) it follows that

$$\|\theta_h^n\|_{L^2(D)} \leq C\frac{t_n}{\Delta t}\min\left(1, \frac{\Delta t\,\|\mathbf{u}\|_{L^\infty((0,T)\times D)^d}}{h}\right) h^{m+1}\,\|c\|_{L^\infty(0,T;H^{m+1}(D))}\,.$$

The result (5.5) follows from this bound together with (5.7). □

Note that $\dfrac{\Delta t\,\|\mathbf{u}\|_{L^\infty((0,T)\times D)^d}}{h}$ is the CFL number, so that according the result (5.5) LG method converges suboptimally with order $O(h^m)$ at low CFL numbers, or in other words, when Δt is much smaller than h; whereas for large CFL numbers, or equivalently, for Δt larger than h, then convergence is of order $O(\dfrac{h^{m+1}}{\Delta t})$.

5.2 Stability and convergence of the modified LG method

Our next concern is to study the stability and convergence of the modified LG method for the linear convection equation. For this purpose we need some auxiliary results in relation to the points $\widetilde{X}^{n,n+1}(x)$. For each j, $1 \leq j \leq NE$, and for each time instant t_n, let us consider the mappings $\widetilde{F}_j^{n,n+1}$ and $F_j^{n,n+1}$.

Lemma 5. *Let $q \geq 1$ be an integer and let $\mathbf{u} \in L^\infty(0,T;W^{q+1,\infty}(D)^d)$. Then $F_j^{n,n+1} \in C^{q,1}(\widehat{T})^d$ and there exists a constant C_1 independent of Δt and h such that for any n*

$$\left\|X^{n,n+1}(x) - \widetilde{X}^{n,n+1}(x)\right\|_{L^\infty(\overline{D})^d} \leq C_1 h^2 \Delta t\,\|\mathbf{u}\|_{L^\infty(0,T;W^{2,\infty}(D)^d)}\,. \tag{5.10}$$

Proof: Recalling the definitions of $X^{n,n+1}(x)$ and $\widetilde{X}^{n,n+1}(x)$, it is convenient to work with the mappings $F_j^{n,n+1}$ and $\widetilde{F}_j^{n,n+1}$ because for each T_j

$$\left\|X^{n,n+1}(x) - \widetilde{X}^{n,n+1}(x)\right\|_{L^\infty(T_j)^d} = \left\|F_j^{n,n+1}(\widehat{x}) - \widetilde{F}_j^{n,n+1}(\widehat{x})\right\|_{L^\infty(\widehat{T})^d}\,. \tag{5.11}$$

First, we show that $F_j^{n,n+1}$ is of class $C^{q,1}(\widehat{T})^d$. Noting that $F_j^{n,n+1}(\widehat{x}) = X^{n,n+1} \circ F_j(\widehat{x})$ and using the integral form of the solution of (2.2) we can write

$$F_j^{n,n+1}(\widehat{x}) = \mathbf{B}_j\widehat{x} + \mathbf{b}_j - \int_{t_n}^{t_{n+1}} \mathbf{u}(X(F_j(\widehat{x}), t_{n+1}; t), t)dt.$$

Since $\mathbf{u} \in L^\infty(0,T;W^{q+1,\infty}(D)^d)$, then applying Lemma 1 it follows that $F_j^{n,n+1}$ is of class $C^{q,1}(\widehat{T})^d$. By virtue of (4.2b) and noticing that $\widehat{I}(\mathbf{B}_j\widehat{x} + \mathbf{b}_j) = \mathbf{B}_j\widehat{x} + \mathbf{b}_j$, the application of Bramble-Hilbert lemma yields

$$\left\|F_j^{n,n+1}(\widehat{x}) - \widetilde{F}_j^{n,n+1}(\widehat{x})\right\|_{L^\infty(\widehat{T})^d} \leq \widehat{C}\,|F_j^{n,n+1}|_{W^{2,\infty}(\widehat{T})^d} =$$

$$\widehat{C}\left|\int_{t_n}^{t_{n+1}} \mathbf{u}(X(F_j(\widehat{x}), t_{n+1}; t), t)dt\right|_{W^{2,\infty}(\widehat{T})^d},$$

where $|\circ|_{W^{2,\infty}(\widehat{T})^d}$ is the semi-norm and $\widehat{C} = C(\widehat{I},\widehat{T})$. In any interval $[t_n, t_{n+1}]$ we have that for each j and $t \in [t_n, t_{n+1}]$, $X(\cdot, t_{n+1}; t) \circ F_j : \widehat{T} \to T_j^{t,n+1} \in \overline{D}$, so that making a change of variable it is easy to see that

$$\left| \int_{t_n}^{t_{n+1}} \mathbf{u}(X(F_j(\widehat{x}), t_{n+1}; t), t) dt \right|_{W^{2,\infty}(\widehat{T})^d} \leq C\Delta t h_j^2 \|\mathbf{u}\|_{L^\infty(t_n,t_{n+1};W^{2,,\infty}(T_j^{*t,n+1})^d)} \leq$$

$$C_1 \Delta t h^2 \|\mathbf{u}\|_{L^\infty(0,T;W^{2,\infty}(D)^d)} .$$

From this inequality and (5.11) the result follows. □

Lemma 6. *Assume the hypotheses of the previous lemma hold true and let* $v \in L^\infty(0, T; H^1(D))$*. At any time* t_n*, there exists a constant* C_2 *independent of* Δt *and* h*, but dependent on* $\|\mathbf{u}\|_{L^\infty(0,T;W^{2,\infty}(D)^d)}$*, such that*

$$\left\| v^n(X^{n,n+1}(x)) - v^n(\widetilde{X}^{n,n+1}(x)) \right\|_{L^2(D)} \leq C_2 h^2 \Delta t \|\nabla v^n\|_{(L^2(D))^d} . \tag{5.12}$$

Proof: Let

$$H_\alpha(x) = \alpha X^{n,n+1} + (1-\alpha)\widetilde{X}^{n,n+1}(x), \quad 0 \leq \alpha \leq 1,$$

then it follows that

$$\left| v^n(X^{n,n+1}(x)) - v^n(\widetilde{X}^{n,n+1}(x)) \right|^2 \leq$$

$$\left| X^{n,n+1}(x) - \widetilde{X}^{n,n+1}(x) \right|^2 \int_0^1 |\nabla v^n(H_\alpha(x))|^2 \, d\alpha.$$

Hence

$$\int_{T_j} \left| v(X^{n,n+1}(x)) - v^n(\widetilde{X}^{n,n+1}(x)) \right|^2 dy \leq$$

$$d^{1/2} \left\| X^{n,n+1}(x) - \widetilde{X}^{n,n+1}(x) \right\|_{L^\infty(\overline{D})^d} \int_0^1 \left(\int_{T_j} |\nabla v^n(H_\alpha(x))|^2 \, dx \right) d\alpha.$$

From (5.10) it follows that

$$\int_{T_j} \left| v^n(X^{n,n+1}(x)) - v^n(\widetilde{X}^{n,n+1}(x)) \right|^2 dx \leq$$

$$Ch^4 \Delta t^2 \|\mathbf{u}\|^2_{L^\infty(0,T;W^{2,\infty}(D)^d)} \|\nabla v^n\|^2_{(L^2(T_j))^d} .$$

Summing with respect to j on both sides we obtain the result. □

Next, we prove a stability result in the L^2-norm for $c_h^n(\widetilde{X}^{n,n+1}(x))$. To do so, we need the inverse inequality

$$\|\nabla v_h\|_{L^2(D)} \leq C h^{-1} \|v_h\|_{L^2(D)} \quad \text{for all } v_h \in W_h. \tag{5.13}$$

This inequality holds true in our meshes D_h because we assume that they are quasi-uniformly regular.

Lemma 7. *There exist constants C_3 and C_4 independent of Δt and h such that for $n = 1, 2, \ldots, N$,*

$$(1 - C_3\Delta t)\|c_h^n\|_{L^2(D)} \leq \left\| c_h^n(\widetilde{X}^{n,n+1}(x)) \right\|_{L^2(D)} \leq (1 + C_4\Delta t)\|c_h^n\|_{L^2(D)}. \tag{5.14}$$

Proof: First, setting $y = X^{n,n+1}(x)$ we have that

$$\int_D \left| c_h^n(X^{n,n+1}(x)) \right|^2 dx = \int_D |c_h^n(y)|^2 \left| (J^{n,n+1})^{-1} \right| dy.$$

From Lemma 2 it follows the existence of constants $K_1 = K_1(C_u)$ and $K_2 = K_2(C_u)$ such that

$$(1 - K_1\Delta t)\|c_h^n\|_{L^2(D)} \leq \left\| c_h^n(X^{n,n+1}(x)) \right\|_{L^2(D)} \leq (1 + K_2\Delta t)\|c_h^n\|_{L^2(D)}. \tag{5.15}$$

Next, noting that

$$c_h^n(\widetilde{X}^{n,n+1}(x)) = c_h^n(X^{n,n+1}(x)) - \left(c_h^n(X^{n,n+1}(x)) - c_h^n(\widetilde{X}^{n,n+1}(x)) \right),$$

it follows by virtue of (5.3) and (5.12) that

$$\left\| c_h^n(\widetilde{X}^{n,n+1}(x)) \right\|_{L^2(D)} \leq (1 + K_2\Delta t)\|c_h^n\|_{L^2(D)} + C_2 h^2 \Delta t \|\nabla c_h^n\|_{(L^2(D))^d}.$$

Using the inverse inequality (5.13) we have that $C_2 h^2 \Delta t \|\nabla c_h^n\|_{(L^2(D))^d} \leq C h \Delta t \|c_h^n\|_{L^2(D)}$; hence, there exists a constant $C_4 = K_2 + hC$ depending on $div\mathbf{u}$, $\|\mathbf{u}\|_{L^\infty(0,T;W^{2,\infty}(D)^d)}$ and the constant of the inverse inequality such that

$$\left\| c_h^n(\widetilde{X}^{n,n+1}(x)) \right\|_{L^2(D)} \leq (1 + C_4\Delta t)\|c_h^n\|_{L^2(D)}.$$

Analogously, setting

$$c_h^n(X^{n,n+1}(x)) = c_h^n(\widetilde{X}^{n,n+1}(x)) - \left(c_h^n(\widetilde{X}^{n,n+1}(x)) - c_h^n(X^{n,n+1}(x)) \right),$$

we obtain that

$$\left\| c_h^n(\widetilde{X}^{n,n+1}(x)) \right\|_{L^2(D)} \geq \left\| c_h^n(X^{n,n+1}(x)) \right\|_{L^2(D)} - \left\| c_h^n(X^{n,n+1}(x)) - c_h^n(\widetilde{X}^{n,n+1}(x)) \right\|.$$

Then, arguing as before we find a constant $C_3 = K_1 + hC$ such that

$$(1 - C_3\Delta t)\|c_h^n\|_{L^2(D)} \leq \left\| c_h^n(\widetilde{X}^{n,n+1}(x)) \right\|_{L^2(D)}.$$

□

We are now in a position to establish the stability in the L^2-norm of the solution calculated by the modified LG method.

Lemma 8. *Under the assumptions of Lemma 7 it follows that for all* $n = 0, 1, 2...$,

$$\left\|c_h^{n+1}\right\|_{L^2(D)} \leq (1 + C_4\Delta t) \left\|c_h^n\right\|_{L^2(D)}. \tag{5.16}$$

Proof: Multiplying by C_i^{n+1} on both sides of (4.3), summing for i, and applying Cauchy-Schwarz inequality it readily follows that

$$\left\|c_h^{n+1}\right\|_{L^2(D)} \leq \left\|c_h^n(\widetilde{X}^{n,n+1}(x))\right\|_{L^2(D)},$$

and from (5.14) the inequality (5.16) follows. □

Now, we proceed to prove the convergence of the method. We have the following result.

Theorem 9. *Let* $c \in C([0,T]; H^{m+1}(D))$ *and the assumptions of Lemma 1 hold. Then there exists a constant* C *independent of* Δt *and* h *such that*

$$\max_n \left\|c(t_n) - c_h^n\right\|_{L^2(D)} \leq Kh^{m+1} \left\|c\right\|_{L^\infty(0,T;H^{m+1}(D))} +$$

$$C\frac{t_n}{\Delta t} \min\left(1, \frac{\Delta t \left\|\mathbf{u}\right\|_{L^\infty([0,T]\times D)^d}}{h}\right) h^{m+1} \left\|c\right\|_{L^\infty(0,T;H^{m+1}(D))} + Ct_n h^2 \left\|\nabla c\right\|_{L^\infty(0,T;L^2(D))}. \tag{5.17}$$

Proof: As in the proof of Theorem 4 we have that for all n the global error $e^n(x) = c^n(x) - c_h^n(x)$ can expressed as

$$e^n(x) = (c^n(x) - P_h c^n(x)) + P_h c^n(x) - c_h^n(x) = \rho^n(x) + \theta_h^n(x).$$

Since $c^n(x) = c^{n-1}(X^{n-1,n}(x)$ and $c_h^n(x) = P_h c_h^{n-1}(\widetilde{X}^{n-1,n}(x))$, then a further decomposition of $P_h c^n(x) - P_h c_h^n(x)$ yields

$$\begin{aligned}\theta_h^n(x) = P_h &\left(c^{n-1}(X^{n-1,n}(x)) - c^{n-1}(\widetilde{X}^{n-1,n}(x))\right) + \\ &P_h\left(c^{n-1}(\widetilde{X}^{n-1,n}(x)) - c_h^{n-1}(\widetilde{X}^{n-1,n}(x))\right).\end{aligned} \tag{5.18}$$

Taking the L^2 norm on both sides of (5.18) and using the contractive property of P_h together with Lemmas 6 and 7 we have that

$$\left\|\theta_h^n\right\|_{L^2(D)} \leq (1 + C_4\Delta t)\left(\left\|\rho^{n-1}\right\|_{L^2(D)} + \left\|\theta_h^{n-1}\right\|_{L^2(D)}\right) +$$

$$C_2 h^2 \Delta t \left\|\nabla c^{n-1}\right\|_{(L^2(D))^d}.$$

Hence, arguing as we did to estimate θ_h^n in the conventional LG method, see (5.8), we have

$$\|\theta_h^n\|_{L^2(D)} \leq Ct_n \frac{h^{m+1}}{\Delta t} \|c\|_{L^\infty(0,T;H^{m+1}(D))} + \\ Ct_n h^2 \|\nabla c\|_{L^\infty(0,T;H^{m+1}(D))}, \tag{5.19}$$

where $C = \max\left((1+C_4\Delta t)Ke^{C_4 t_n},\ C_2 e^{C_4 t_n}\right)$. This estimate, as it happens in the conventional LG method, breaks down when $\Delta t \to 0$ for a fixed h. To get an estimate when Δt is much smaller than h we argue in the same fashion as in the conventional LG method; thus, returning to (5.18), we can write

$$\theta_h^n(x) = P_h\left(c^{n-1}(X^{n-1,n}(x)) - c^{n-1}(\widetilde{X}^{n-1,n}(x))\right) + \\ \theta_h^{n-1}(\widetilde{X}^{n-1,n}(x)) + P_h\left(\rho^{n-1}\circ X^{n-1,n}(x) - \rho^{n-1}(x)\right).$$

Taking L^2-norm on both sides of this inequality and using the same argument as above, we get now that

$$\|\theta_h^n\|_{L^2(D)} \leq (1+C_4\Delta t)\left\|\theta_h^{n-1}\right\|_{L^2(D)} + C_2 h^2 \Delta t \left\|\nabla c^{n-1}\right\|_{(L^2(D))^d} + \\ \left\|\rho^{n-1}\circ X^{n-1,n}(x) - \rho^{n-1}(x)\right\|_{L^2(D)}.$$

Hence, arguing similarly as we did to obtain (5.9) we have that

$$\|\theta_h^n\|_{L^2(D)} \leq C\tfrac{t_n}{\Delta t}\left(\frac{\Delta t\,\|\mathbf{u}\|_{L^\infty((0,T)\times D)^d}}{h}\right) h^{m+1}\|c\|_{L^\infty(0,T;H^{m+1}(D))} + \\ Ct_n h^2 \|\nabla c\|_{L^\infty(0,T;H^{m+1}(D))}, \tag{5.20}$$

where $C = \max\left((1+C_4\Delta t)Ke^{C_4 t_n}, C_2 e^{C_4 t_n}\right)$. From (5.19), (5.20) and (5.7) the result follows. □

6 Numerical tests

To test the performance of the LG methods we run typical model problems of different degrees of difficulty, ranging from smooth solutions everywhere to solutions with strong discontinuities. The numerical examples have been run using P_2 and P_1 elements, though we shall only present the results corresponding to P_2 elements because the results with P_1 elements yield similar conclusions.

6.1 Example 1. Rotating cylinder

The first example consists of simulating the time evolution of an initial condition represented by a circular cylinder of radius $\frac{1}{4}$, in a fixed rotating velocity field defined on an elliptical domain D of semi-axes $a = 1.5$ and $b = 1.0$. The initial value problem is:

$$\begin{cases} \dfrac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = 0 \quad \text{in } D \times (0, T], \\ c^0(x) = \begin{cases} 1, & \text{if } x \in (x_1 - 0.75)^2 + x_2^2 \leq \frac{1}{16}, \\ 0, & \text{otherwise,} \end{cases} \end{cases}$$

with velocity $\mathbf{u} = (-2x_2, x_1)$. Note that the exact solution of this problem is the initial condition transported by the velocity field along the trajectories, which are ellipses, thus

$$c(x, t) = c^0(x^0)$$

with

$$\begin{cases} x_1(t) = x_1^0 \cos(\sqrt{2}t) + x_2^0 \frac{\sqrt{2}}{2} \sin(\sqrt{2}t), \\ x_2(t) = -x_1^0 \sqrt{2} \sin(\sqrt{2}t) + x_2^0 \cos(\sqrt{2}t). \end{cases} \tag{6.1}$$

This means that the initial cylinder, as it rotates, changes the shape from circular, when $\sqrt{2}t = n\pi$, $n = 0, 1, 2, ..,$ to elliptical, when $\sqrt{2}t = \frac{\pi}{2} + n\pi$, $n = 0, 1, ...$ Note that the solution changes its shape with time, but the triangles keep their sides straight, though they stretch and shorten maintaining constant the values of their areas, because the velocity field is a time independent linear function of the space coordinates.

We now present the numerical results obtained with P_2 elements. The results of the modified LG method (SLG in the figures) are compared with those obtained by the conventional LG method (in both methods the integrals are approximated by Hammer's quadrature rule of 7 points) and the semi-Lagrangian (SL) scheme, which uses interpolatory projection instead of the L^2 projection. The interpolatory projection is performed with finite element interpolation. Figure 2 shows the evolution in time of $\int c_h^n dx$, $\int (c_h^n)^2 dx$, the L^2-norm error and the profile of the cylinder at the cross-section $x_2 = 0$ after 5 revolutions. In the cross-section, the profile of the exact solution is represented by the dashed line while the profiles of the numerical approximations are represented by continuous lines as indicated in the figure. It is clear from the panels of this figure that the behavior of the modified LG method is practically the same as the one of the conventional LG method, and both perform better than the SL scheme with quadratic interpolation. It is worth noticing that the errors of the different numerical methods experience an exponential increase at the first few time steps and then they grow much more slowly, although the slope of the SL error is higher than the slope of the modified LG and conventional LG errors.

We show in Table 1 the variation of the L^2-norm error as a function of h, while maintaining constant the time step $\Delta t = \frac{\sqrt{2}\pi}{80}$, and in Table 2 the L^2-norm error in terms of the variation of the time step Δt when the numerical solution is calculated in a very fine mesh.

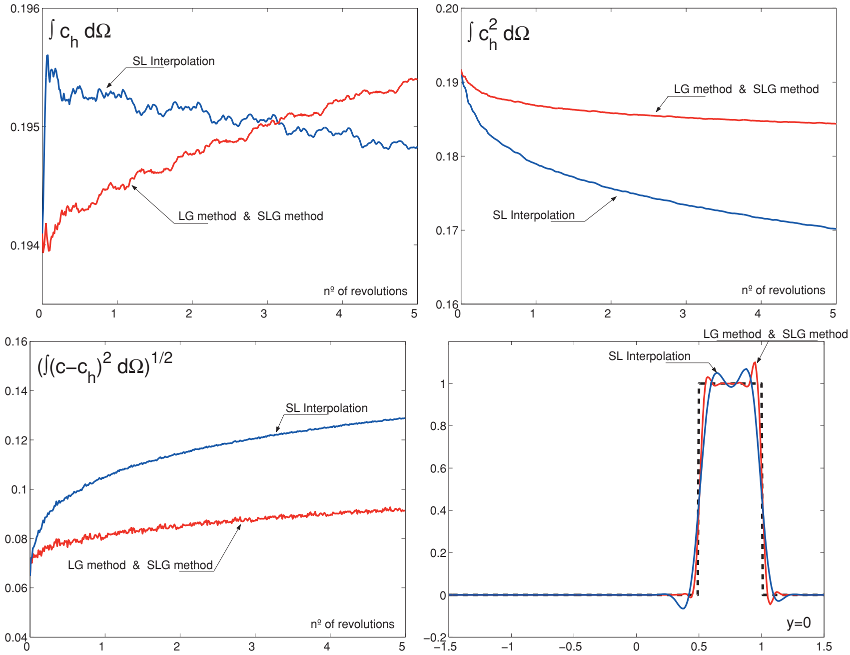

Figure 2: Results with P_2 elements for the Example 1.

$\left[\int (c - c_h)^2\, dx\right]^{1/2}$	*SL Interpolation*	*LG method*	*SLG method*
N *elements* $= 1.606$	0.184	0.132	0.132
N *elements* $= 6.424$	0.128	0.091	0.091
N *elements* $= 25.696$	0.090	0.064	0.064

Table 1: L^2-error after 400 time steps as function of h when $\Delta t = \frac{\sqrt{2}\pi}{80}$ in a mesh of quadratic triangles for the Example 1.

In both tables the errors are calculated after 5 revolutions. We must say that in all the results presented above, the feet of the characteristics have been calculated using the analytical velocity in order to remove the error committed in the case they were approximated by solving (2.2) numerically, as it should be done in a more general setting because, in general, the velocity is known at the mesh points and in a discrete set of time instants $\{t_n\}$. This numerical approximation would yield an error that has to be added to the error of the method estimated in the previous section. However, to illustrate the influence that the approximation of the feet of the characteristics has on the global error of the method,

$\left[\int (\mathbf{c}-\mathbf{c_h})^2\,\mathbf{dx}\right]^{\mathbf{1/2}}$	*SL Interpolation*	*LG method*	**SLG method**
$\Delta t = \sqrt{2}\pi/40$	0.082	0.059	**0.059**
$\Delta t = \sqrt{2}\pi/80$	0.090	0.064	**0.064**
$\Delta t = \sqrt{2}\pi/160$	0.101	0.068	**0.068**

Table 2: L^2-error after 5 revolutions as function of Δt in a mesh of 25696 quadratic triangles for the Example 1 when the feet of the characteristics are calculated exactly. The first row results correspond to 200 time steps, the second row results to 400 time steps and the third row results to 800 time steps.

we repeat the experiments of Table 2 but when the feet of the characteristics are calculated by solving (2.2) numerically with a second order method. These results are shown in Table 3.

$\left[\int (\mathbf{c}-\mathbf{c_h})^2\,\mathbf{dx}\right]^{\mathbf{1/2}}$	*SL Interpolation*	*LG method*	**SLG method**
$\Delta t = \sqrt{2}\pi/40$	0.155	0.170	**0.170**
$\Delta t = \sqrt{2}\pi/80$	0.095	0.076	**0.076**
$\Delta t = \sqrt{2}\pi/160$	0.101	0.068	**0.068**

Table 3: L^2-error after 5 revolutions as function of Δt in a mesh of 25696 quadratic triangles for the Example 1 when the feet of the characteristics are approximated by a second order in time numerical method.

By comparing Tables 2 and 3 we see that for Δt sufficiently small ($\Delta t = \frac{\sqrt{2}\pi}{80}$ in this example) the approximation of the feet of the characteristics by a second order method has a small influence on the global error.

CPU time 5 rev	*SL Interpolation*	*LG method*	**SLG method**
analytical velocity	20.3	71.2	**58.1**
velocity at the nodes of the mesh at every time step	376.6	681.7	**206.3**

Table 4: CPU time in seconds for 5 revolutions in a mesh of 6424 quadratic triangles and $\Delta t = \frac{\sqrt{2}\pi}{80}$ for the Example 1.

Now, we evaluate the CPU time needed for each method to calculate the solution. First of all, we must say that although for this example equation (2.2) is an autonomous system, so that we could have calculated the feet of the characteristics once and for all at the first time step, we have chosen to calculate them at every time step considering that in a more general problem, for example when the velocity is also time dependent, the feet of the characteristics must be calculated every time step. Table 4 shows the CPU time spent for each method to calculate the solution after 5 revolutions in a mesh of 6424 triangles with a time step $\Delta t = \frac{\sqrt{2}\pi}{80}$. In the first row it is shown the CPU time when the feet of the characteristics are calculated using the analytical velocity, in the second row the CPU time corresponds to the case when the feet are calculated by the second order method of Algorithm 3 with the values of the velocity known at the mesh points and time instants t_n, t_{n-1}. In both cases, either analytical formula or numerical method, it is necessary to use a search-locate algorithm to identify the triangles where the feet of the characteristics are located at each time instant t_n. For this purpose, we use the search-locate algorithm of Allievi and Bermejo (1997) specifically designed for unstructured meshes.

According to the results of this table, it is clear that the methods spent most of the CPU time in calculating (approximating) and locating the departure points. This is the reason why the modified LG method performs better than the conventional LG method as far as the expenditure of CPU time concerns; for we should recall that in the modified LG method we only need to calculate the feet of the characteristics corresponding to the vertices of the elements, whereas in the conventional LG method one has to compute for each element of the mesh the feet of the characteristics that correspond to quadrature points. Moreover, and at least for this example, we also notice that the modified LG method may perform better, in terms of CPU time, than the conventional SL with quadratic interpolation, because for the latter method the number of feet of the characteristics we need to calculate at each time step is near twice as much as for the modified LG method. Clearly, the LG methods are more accurate than the conventional SL finite element interpolation method.

6.2 Example 2. Rotating Gaussian bell

This example, with a sufficiently smooth initial condition, has been chosen with the main purpose of testing the error estimate of Theorems 4 and 9. Thus, with both the same domain D and the same rotating velocity, $u = (-2x_2, x_1)$ of the previous example, we consider the initial condition

$$c^0(x) = \exp\left(-25\left[\left(x_1 - \frac{3}{4}\right)^2 + x_2^2\right]\right)$$

that represents a Gaussian bell with center at $(0.75, 0)$. We present results of this test with P_2 elements. The quadrature rule for either modified or conventional LG methods, in both P_2 and P_1 elements, is Hammer 7 points.

We show in Figure 3 the time evolution of $\int c_h^n dx$, $\int (c_h^n)^2 dx$, the L^2-norm error and the profile of the Gaussian bell at the cross section $x_2 = 0$ after 5 revolutions. The exact solution is represented by the dashed line whereas the numerical solutions are represented by the full lines as indicated in the figure of the cross-section. It is clear from the figure that the behavior of the modified LG scheme is practically the same as the conventional LG method, and both perform better than the SL scheme with quadratic interpolation. This fact is also confirmed by Tables 5 and 6, where, in Table 5, we show the variation of

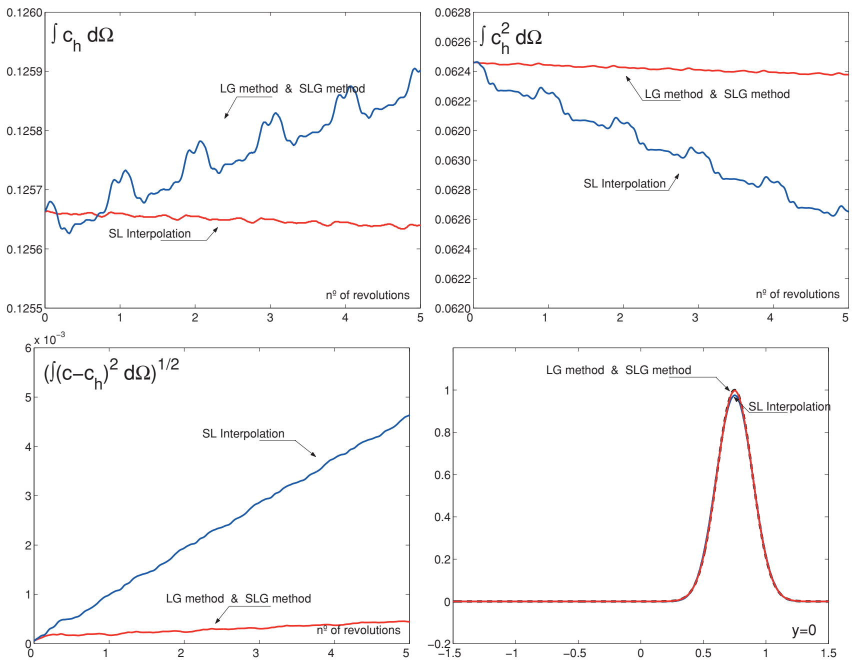

Figure 3: Results with P_2 elements for the Example 2.

$\left[\int (\mathbf{c}-\mathbf{c_h})^2\, \mathbf{dx}\right]^{\mathbf{1/2}}$	*SL Interpolation*	*LG method*	**SLG method**
N elements $= 1.606$	$4.53 \cdot 10^{-2}$	$3.71 \cdot 10^{-3}$	$\mathbf{3.71 \cdot 10^{-3}}$
N elements $= 6.424$	$4.63 \cdot 10^{-3}$	$4.43 \cdot 10^{-4}$	$\mathbf{4.43 \cdot 10^{-4}}$
N elements $= 25.696$	$3.24 \cdot 10^{-4}$	$4.71 \cdot 10^{-5}$	$\mathbf{4.71 \cdot 10^{-5}}$

Table 5: L^2-error as function of h when $\Delta t = \frac{\sqrt{2}\pi}{80}$ in a mesh of quadratic triangles for the Example 2.

the L^2-norm error as a function of h while maintaining constant the time step $\Delta t = \frac{\sqrt{2}\pi}{80}$, and in Table 6 the L^2-norm error in terms of the variation of the time step Δt when the numerical solution is calculated in a very fine mesh. We must say that in all the results presented above, the feet of the characteristics have been calculated using the analytical velocity. Since for this example the velocity vector is a linear function that does not depend on time, then the triangles $T_j^{n,n+1}$ are straight, so that $F_j^n(\widehat{x}) = \widetilde{F}_j^n(\widehat{x})$, and, therefore, $\left\| X^{n,n+1}(x) - \widetilde{X}^{n,n+1}(x)) \right\|_{L^\infty(\mathbb{R}^2)^2} = 0$; this means that the term h^2 is absent in the error estimate. By inspection of Table 5 we notice the following: (1) modified LG error= conventional LG error, and this error is much smaller than SL error. (2) When we move down from h_1 (first row in the table) to $h_2 = h_1/2$ (second row) and to $h_3 = h_1/4$ (third row) we see that the errors are $e_2 < (h_2/h_1)^3 e_1$ and $e_3 < (h_3/h_2)^3 e_2$, e_i being the error of i-th row after 5 revolutions. Since $h_1 = 0.1284$, $h_2 = 0.0640$, $h_3 = 0.032$ and $\Delta t = \frac{\sqrt{2}\pi}{80} = 0.0555$, these results confirm the estimate of Theorems 4 and 9. Turning the attention to Table 6, we notice that the error (in each method) increases when Δt decreases as predicted by the error analysis. An optimal error estimate Δt should be of the order of h.

$\left[\int (\mathbf{c} - \mathbf{c_h})^2 \, \mathbf{dx}\right]^{1/2}$	*SL Interpolation*	*LG method*	**SLG method**
$\Delta t = \sqrt{2}\pi/40$	$1.44 \cdot 10^{-4}$	$4.75 \cdot 10^{-5}$	$\mathbf{4.75 \cdot 10^{-5}}$
$\Delta t = \sqrt{2}\pi/80$	$3.24 \cdot 10^{-4}$	$4.71 \cdot 10^{-5}$	$\mathbf{4.71 \cdot 10^{-5}}$
$\Delta t = \sqrt{2}\pi/160$	$9.25 \cdot 10^{-4}$	$1.05 \cdot 10^{-4}$	$\mathbf{1.05 \cdot 10^{-4}}$

Table 6: L^2-error as function of Δt in a mesh of 25696 quadratic triangles for Example 2.

PART II

LG method for convection-diffusion equations

7 The continuum problem

We shall study second order in time LG methods to calculate the numerical solution of convection-diffusion equations. Specifically, we consider the model problem

$$\left\{ \begin{array}{c} \dfrac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = \nabla \cdot (\nu(x,t)\nabla c) + f(x,t) \ \text{ in } \ D \times (0,T), \\ c(x,t) = 0, \quad (x,t) \in \partial D \times (0,T), \\ c(x,0) = c^0(x), \quad x \in D, \end{array} \right. \tag{7.1}$$

where D is an open bounded domain of $\mathbb{R}^d$ ($d = 2$ or 3) with a Lipschitz continuous boundary ∂D, $c : \overline{D} \times [0,T) \to \mathbb{R}$, $\mathbf{u} : D \times (0,T) \to \mathbb{R}^d$ is a vector-valued function that represents a flow velocity and $\nu(x,t)$ is a symmetric positive definite matrix of diffusion coefficients such that for all t, the ratio $\varkappa = \dfrac{\lambda_{\max}}{\lambda_{\min}}$ is moderate, with $\lambda_{\max}$ and $\lambda_{\min}$ denoting the largest and smallest eigenvalues of $\nu(x,t)$ respectively. $\mathbf{u}(x,t) \in L^\infty(D \times (0,T])$, $|\mathbf{u}(x,t)| \gg \lambda_{\min}$, and for simplicity we shall consider that $\mathbf{u}(x,t) = \mathbf{0}$ on $(x,t) \in \partial D \times (0,T]$. If in addition, we assume that the coefficients $\nu_{ij}(x,t)$ of the matrix $\nu(x,t)$ are in $L^\infty(D \times (0,T])$, $f \in L^2(D \times (0,T])$ and $c^0 \in L^2(D)$, then it can be shown that (7.1) has a unique weak solution $c \in L^2(0,T; H_0^1(D)) \cap C([0,T]; L^2(D))$, $\dfrac{\partial c}{\partial t} \in L^2(0,T; H^{-1}(D))$ that satisfies for each $v \in H_0^1(D)$ a.e. time $0 \leq t \leq T$,

$$\begin{gathered} \left\langle \frac{Dc}{Dt}, v \right\rangle + a\,(t; c, v) = (f, v), \\ c(x,0) = c^0(x), \end{gathered} \tag{7.2}$$

where $a\,(t; \cdot, \cdot) : H_0^1(D) \times H_0^1(D) \to \mathbb{R}$ is a continuous coercive bilinear form defined as

$$a\,(t; c, v) = \int_D \nu(x,t) \nabla c \cdot \nabla v dx, \tag{7.3}$$

$\langle \cdot, \cdot \rangle$ denotes the duality pairing for H_0^1 and its dual H^{-1}, and $(\cdot, \cdot)$ is the usual inner product in $L^2(D)$. To calculate numerically the weak solution by LG methods, the interval $[0,T]$ is divided into subintervals $[t_{n-1}, t_n]$, $n = 1, 2, \cdots, N$, of length $\Delta t_n = t_n - t_{n-1}$ such that $0 = t_0 < t_1 < \cdots < t_N = T$, and the total derivative $\dfrac{Dc}{Dt}$ is discretized along the characteristic curves $X(x, t_{n+1}; t)$. To motivate the LG methods we shall study to solve (7.1), we consider for each interval $[t_n, t_{n+1}]$ the following mappings. (1) For $x \in \overline{D}$, and $t \in [t_n, t_{n+1}]$, $x \to X(x, t_{n+1}, t)$, where $X(x, t_{n+1}, t)$ is the unique solution of (2.2), and it is proven that $X(x,t_{n+1};t) \in \overline{D}$ because the velocity $\mathbf{u}$ vanishes on Γ. (2) Setting $y = X(x,t_{n+1},t)$, the inverse map of X is the mapping $y \to Y(y,t;t_{n+1}) = x$ with the condition $Y(y,t_n;t_{n+1}) = y$. Note that $\dfrac{dY(y,t;t_{n+1})}{dt} = \mathbf{u}(x,t)$. Let $F(x,t_{n+1};t) = \left(\dfrac{\partial X}{\partial x}\right)$ and J be its determinant, we have that $F(x,t_{n+1};t_{n+1}) = I$ (the unit matrix) and $J = 1$.

Introducing the change of variables $c(x,t) = c(Y(y,t;t_{n+1}),t) = \overline{c}(y,t)$ we can easily compute, see [3],

$$\frac{\partial \overline{c}(y,t)}{\partial t} = \frac{\partial c}{\partial t} + \mathbf{u}\cdot\nabla_x c,\;\; \nabla_x c = F^{-T}\nabla_y \overline{c},$$

$$div_x(\nu(x,t)\nabla_x c) = \frac{1}{J} div_y\,(JF^{-1}\overline{\nu}(y,t)\,F^{-T}\nabla_y\overline{c}),$$

so that the partial differential equation of (7.1) can be recast as

$$\frac{\partial \overline{c}(y,t)}{\partial t} - \frac{1}{J} div_y\left(JF^{-1}\overline{\nu}(y,t)\right) F^{-T}\nabla_y\overline{c}) = \overline{f}(y,t) \text{ in } D\times(t_n,t_{n+1}]\,, \tag{7.4a}$$

or equivalently

$$\frac{\partial \overline{c}(y,t)}{\partial t} = G(y,t) \text{ in } D\times(t_n,t_{n+1}];\; G(y,t) = \frac{1}{J} div_y\left(JF^{-1}\overline{\nu}(y,t)\right) F^{-T}\nabla_y\overline{c}) = \overline{f}(y,t). \tag{7.4b}$$

Discretizing in time (7.4b) by the Backwards Differentiation Formula of order 1 (BDF1) it follows the equation

$$\frac{\overline{c}(y,t_{n+1}) - \overline{c}(y,t_n)}{\Delta t} = G(y,t_{n+1});$$

noting that (1) for $t = t_{n+1}$, $\overline{c}(y,t_{n+1}) = c(Y(y,t_{n+1};t_{n+1}),t_{n+1}) = c(x,t_{n+1})$, $J = 1$, $F = I$, $\overline{f}(y,t_{n+1}) = f(x,t_{n+1})$, and (2) for $t = t_n$, $\overline{c}(y,t_n) = c(Y(y,t_n;t_{n+1}),t_n) = c(X(x,t_{n+1};t_n),t_n)$, it follows that

$$\frac{c(x,t_{n+1}) - c(X(x,t_{n+1};t_n),t_n)}{\Delta t} = div_x(\nu(x,t_{n+1})\nabla_x c(x,t_{n+1}) + f(x,t_{n+1})\,. \tag{7.5}$$

This is the first order in time LG method proposed by Douglas and Russell (1982) and Pironneau (1982). A second order scheme proposed by Ewing and Russell (1981) is obtained by discretizing in time (7.4b) by the BDF of order 2 (BDF2), thus we have

$$\frac{3\overline{c}(y,t_{n+1}) - 4\overline{c}(y,t_n) + \overline{c}(y,t_{n-1})}{2\Delta t} = G(y,t_{n+1}),$$

or equivalently

$$\begin{gathered}\frac{3c(x,t_{n+1}) - 4c(X(x,t_{n+1};t_n),t_n) + c(X(x,t_{n+1};t_{n-1}),t_{n-1})}{2\Delta t} = \\ div_x(\nu(x,t_{n+1})\nabla_x c(x,t_{n+1}) + f(x,t_{n+1}).\end{gathered} \tag{7.6}$$

Another second order scheme proposed by Bermúdez et al. (2006) is obtained by discretizing (7.4b) by the trapezoidal rule:

$$\frac{\overline{c}(y,t_{n+1})-\overline{c}(y,t_n)}{\Delta t}=\frac{1}{2}\left(G(y,t_{n+1})+G(y,t_n)\right), \tag{7.7}$$

this yields

$$\begin{aligned}\frac{c(x,t_{n+1})-c(X(x,t_{n+1};t_n),t_n)}{\Delta t}&=\frac{1}{2}div_x(\nu(x,t_{n+1})\nabla_x c(x,t_{n+1})+\\ \frac{1}{J}\frac{1}{2}div_y\left(JF^{-1}\overline{\nu}(y,t_n)\,F^{-T}\nabla_y\overline{c}(y,t_n)\right.&+\frac{1}{2}(f(x,t_{n+1})+f(X(x,t_{n+1};t_n),t_n)\end{aligned} \tag{7.8}$$

For the same accuracy, the latter scheme is more computationally expensive than (7.6) because the Jacobians at the points $X(x,t_{n+1};t_n)$ plus the calculation of the right hand side integrals have to be performed every time step. Another second order scheme of the same type has also been proposed by Rui and Tabata (2002). In what follows we shall focus on the LG-BDF2 methods because in terms of accuracy versus CPU time and computer storage they are better than the other ones.

8 Auxiliary results

We recall in this section some results concerning the approximation properties of the finite element spaces and the convergence properties of the operators $I_h : C(D) \rightarrow W_h$ and $R_h : H_0^1(D) \rightarrow V_h$. In the error estimates there will be multiplying constants, independent of the mesh parameter, that we denote by C and are different from place to place

Let h be the mesh parameter such that given h_0 sufficiently small $0 < h \leq h_0 < 1$, then there exists a constant C such that for $u \in H^{s+1}(D) \cap H_0^1(D),\ 1 \leq s \leq m$,

$$\inf_{v_h\in V_h}\left\{\|u-v_h\|_{L^2(D)}+h\,\|\nabla\,(u-v_h)\|_{L^2(D)}\right\}\leq Ch^{s+1}\,\|u\|_{H^{s+1}(D)}\,. \tag{8.1}$$

For the Lagrange interpolation operator $I_h : C(D) \rightarrow W_h$ we have that for $u \in H^{s+1}(D),\ s > 1$,

$$\|u-I_h u\|_{L^2(D)}+h\,\|\nabla\,(u-I_h u)\|_{L^2(D)}\leq Ch^{s+1}\,\|u\|_{H^{s+1}(D)}\,. \tag{8.2}$$

Finally, we shall consider the elliptic projector operator $R_h : H_0^1 \rightarrow V_h$ defined in relation with the bilinear form $a(t;u,v)$, then for $u \in L^\infty(0,T;H_0^1(D) \cap H^{s+1}(D))$ with $u(0) \in H_0^1(D) \cap H^{s+1}(D)$, $R_h u$ is the solution of the problem

$$a(t;R_h u,v_h)=a(t,u,v_h),\quad \text{for all } v_h\in V_h. \tag{8.3}$$

Let $\rho := u - R_h u$, then under proper regularity assumptions the following estimates hold:

$$\|\rho\|_{L^2(D)}+h\,\|\nabla\rho\|_{L^2(D)}\leq Ch^{s+1}\,\|u\|_{H^{s+1}(D)} \tag{8.4a}$$

and

$$\|\rho_t\|_{L^2(D)} + h\,\|\nabla\rho_t\|_{L^2(D)} \leq Ch^{s+1}\left[\|u\|_{H^{s+1}(D)} + \|u_t\|_{H^{s+1}(D)}\right], \tag{8.4b}$$

where ρ_t stands for $\dfrac{\partial\rho}{\partial t}$ and the constant C in (8.4a) and (8.4b) is of the form $C = \dfrac{\|\nu\|_{L^\infty(D\times(0,T))}}{\nu_0} \times C_D$, with C_D being a generic positive approximation constant.

9 The conventional and modified LG-BDF2 methods

LG-BDF2 method

For $n = 1, \ldots, N-1$, given c_h^n and c_h^{n-1}, find $c_h^{n+1} \in V_h$ such that for all $v_h \in V_h$

$$\begin{aligned}\frac{1}{2}\left(3c_h^{n+1} - 4c_h^n(X^{n,n+1}(x)) + c_h^{n-1}(X^{n-1,n+1}(x)), v_h\right) +\\ \Delta t a^{n+1}(c_h^{n+1}, v_h) = \Delta t(f^{n+1}, v_h),\end{aligned} \tag{9.1a}$$

and for $n = 1$, let $c_h^0 = R_h c^0(x) \in V_h$, find $c_h^1 \in V_h$ such that for all $v_h \in V_h$

$$\left(c_h^1 - c_h^0(X^{0,1}(x)), v_h\right) + \Delta t a^1(c_h^1, v_h) = (f^1, v_h). \tag{9.1b}$$

Here $a^{n+1}(c_h^{n+1}, v_h)$ and $a^1(c_h^1, v_h)$ are shorthand notations for the bilinear forms $a(t_{n+1}; c_h^{n+1}, v_h)$ and $a(t_1; c_h^1, v_h)$ respectively.

Remark 11. *Noting that $X^{n-1,n+1}(x)$ is the shorthand notation for $X(x, t_{n+1}; t_{n-1})$, which is solution of (2.2) at time t_{n-1}, then it is easy to see, setting $y = X^{n,n+1}(x)$, that $X^{n-1,n+1}(x) = X^{n-1,n} \circ X^{n,n+1}(x) = X^{n-1,n}(y)$.*

Remark 12. *The calculation of c_h^{n+1} by (9.1a) requires the solution of a symmetric linear system, which, in general, is very well conditioned. This is one of the properties that make appealing LG methods for convection diffusion equations.*

The calculation of c_h^{n+1} by the conventional LG-BDF2 method consists of the following steps:

Step 1 Calculate the integrals $(c_h^n(X^{n,n+1}(x), v_h)$ and $(c_h^n(X^{n-1,n+1}(x), v_h)$ by using Algorithm 1 of Part 1

Step 2 Calculate the term $a^{n+1}(c_h^{n+1}, v_h)$ to produce the stiffness $\mathbf{S}$. If the diffusion ν were independent of t, this step would be done once and for all at the initial step.

Step 3 Solve the system

$$(3\mathbf{M}{+}2\Delta t\mathbf{S})\left[\mathbf{C}^{n+1}\right] = \mathbf{R}^n.$$

As we have studied in Part I, a crucial step of these methods is the computation of the terms $(c_h^n(X^{n,n+1}(x)), v_h)$ and $\left(c_h^{n-1}(X^{n-1,n+1}(x)), v_h\right)$, this is the reason why we have devised the modified LG-BDF2 methods in order to calculate such terms in a more efficient manner, but without injuring the order of convergence of the conventional LG methods.

Modified LG-BDF2 method

For $n = 1, \ldots, N-1$*, given* c_h^n *and* $c_h^{n-1} \in V_h$

$$\frac{1}{2}\left(3c_h^{n+1} - 4c_h^n(\widetilde{X}^{n,n+1}(x)) + 2c_h^{n-1}(\widetilde{X}^{n-1,n+1}(x)), v_h\right) + \\ \Delta t a^{n+1}(c_h^{n+1}, v_h) = \Delta t (f^{n+1}, v_h), \tag{9.2a}$$

and for $n = 1$*, given* $c_h^0 = R_h c^0$*, find* $c_h^1 \in V_h$ *such that for all* $v_h \in V_h$

$$\left(c_h^1 - c_h^0(\widetilde{X}^0(x)), v_h\right) + \Delta t a^1(c_h^1, v_h) = (f^1, v_h). \tag{9.2b}$$

The only difference about the implementation of conventional and modified LG-BDF2 methods is in Step 1. In the modified LG-BDF2 method this step is performed by using Algorithm 2 of Part I.

10 Analysis of the LG methods for linear convection-diffusion equations

In this section we study the convergence of the LG methods presented in these notes for convection-diffusion problems. We describe a general methodology which allows us to show the existence of a convergence regime depending on both the character of the equations and the magnitude of the CFL number. If the problem is diffusion dominated, then the convergence of LG methods in the L^2-norm is optimal; however if the character is convection dominated, then at small CFL numbers the space convergence of these methods is suboptimal.

To keep the limits of the length of these notes, we shall only analyze the modified LG methods because their analysis is more involved, however, the same technique, with obvious changes, is applied to the conventional LG methods.

First, we establish a stability result.

Lemma 10. *Assuming that* $\mathbf{u} \in L^\infty(0,T; W^{2,\infty}(D)^d)$*, then there exists a positive constant* K *independent of* Δt *and* h *such that for all* $n = 1, 2, \ldots, N$*, the solution* c_h^{n+1} *calculated by* (9.2a) *satisfies*

$$\|c_h^N\|_{L^2(D)} \leq K\left(\|c_h^0\|_{L^2(D)} + 2\Delta t \sum_{n=1}^{N} \|f^n\|_{L^2(D)}\right). \tag{10.1}$$

Proof: To simplify the writing of the formulae that follow we introduce further notation: let $b^n(\cdot)$ denote a generic function at time instant t_n, then we set

$$b^n = b^n(x),\ \ \widetilde{b}^{*n} = b^n(\widetilde{X}^{n,n+1}(x)),\ \ b^{*n} = b^n(X^{n,n+1}(x)),$$

$$\widetilde{b}^{**n-1} = b^{n-1}(\widetilde{X}^{n-1,n+1}(x)),\ \ b^{**n-1} = b^{n-1}(X^{n-1,n+1}(x)),$$

$$\overline{\Delta}_1 b^{n+1} = b^{n+1} - b^{*n} \ \text{ and } \ \overline{\Delta}_2 b^{n+1} = b^{n+1} - b^{**n-1}.$$

Then, recasting the first term of (9.2a) as

$$\frac{1}{2}\left(3c_h^{n+1} - 4\widetilde{c}_h^{*n} + \widetilde{c}_h^{**n-1}, v_h\right) = \frac{1}{2}\left(3c_h^{n+1} - 4c_h^{*n} + c_h^{**n-1}, v_h\right) -$$

$$2(\widetilde{c}_h^{*n} - c_h^{*n}, v_h) + \frac{1}{2}(\widetilde{c}_h^{**n-1} - c_h^{**n-1}, v_h) =$$

$$\left(2\overline{\Delta}_1 c_h^{n+1} - \frac{1}{2}\overline{\Delta}_2 c_h^{n+1}, v_h\right) -$$

$$2(\widetilde{c}_h^{*n} - c_h^{*n}, v_h) + \frac{1}{2}(\widetilde{c}_h^{**n-1} - c_h^{**n-1}, v_h),$$

and using the relations

$$2(\overline{\Delta}_l c_h^{n+1}, c_h^{n+1}) = \overline{\Delta}_l \left\| c_h^{n+1} \right\|^2_{L^2(D)} + \left\| \overline{\Delta}_l c_h^{n+1} \right\|^2_{L^2(D)} \quad \text{for } l = 1 \text{ or } 2,$$

it follows by setting $v_h = c_h^{n+1}$ in (9.2a) that

$$\begin{aligned} &\overline{\Delta}_1 \left\| c_h^{n+1} \right\|^2_{L^2(D)} + \left\| \overline{\Delta}_1 c_h^{n+1} \right\|^2_{L^2(D)} - \tfrac{1}{4}\overline{\Delta}_2 \left\| c_h^{n+1} \right\|^2_{L^2(D)} - \\ &\tfrac{1}{4} \left\| \overline{\Delta}_2 c_h^{n+1} \right\|^2_{L^2(D)} + \Delta t a^{n+1}(\nabla c_h^{n+1}, \nabla c_h^{n+1}) = \\ &\Delta t(f_h^{n+1}, c_h^{n+1}) + 2(\widetilde{c}_h^{*n} - c_h^{*n}, c_h^{n+1}) - \frac{1}{2}(\widetilde{c}_h^{**n-1} - c_h^{**n-1}, c_h^{n+1}). \end{aligned} \tag{10.2}$$

Next, we bound the terms $\overline{\Delta}_1 \left\| c_h^{n+1} \right\|^2_{L^2(D)}$, $\overline{\Delta}_2 \left\| c_h^{n+1} \right\|^2_{L^2(D)}$ and $\left\| \overline{\Delta}_2 c_h^{n+1} \right\|^2_{L^2(D)}$. Making use of (5.15) it is easy to show that

$$\overline{\Delta}_1 \left\| c_h^{n+1} \right\|^2_{L^2(D)} \geq \Delta_1 \left\| c_h^{n+1} \right\|^2_{L^2(D)} - K_2 \Delta t \left\| c_h^{n} \right\|^2_{L^2(D)};$$

similarly,

$$\overline{\Delta}_2 \left\| c_h^{n+1} \right\|^2_{L^2(D)} \leq \Delta_2 \left\| c_h^{n+1} \right\|^2_{L^2(D)} + 2K_1 \Delta t \left\| c_h^{n-1} \right\|^2_{L^2(D)},$$

then, substituting these inequalities in (10.2) we obtain that

$$\Delta_1 \left\| c_h^{n+1} \right\|^2_{L^2(D)} - \frac{1}{4}\Delta_2 \left\| c_h^{n+1} \right\|^2_{L^2(D)} + \left\| \overline{\Delta}_1 c_h^{n+1} \right\|^2_{L^2(D)} - \frac{1}{4} \left\| \overline{\Delta}_2 c_h^{n+1} \right\|^2_{L^2(D)} +$$
$$\Delta t a^{n+1}(\nabla c_h^{n+1}, \nabla c_h^{n+1}) \leq \Delta t (f_h^{n+1}, c_h^{n+1}) + \overline{K}_{12}\Delta t \left(\|c_h^n\|^2_{L^2(D)} + \left\| c_h^{n-1} \right\|^2_{L^2(D)} \right) + \quad (10.3)$$
$$2(\widetilde{c}_h^{*n} - c_h^{*n}, c_h^{n+1}) - \frac{1}{2}(\widetilde{c}_h^{**n-1} - c_h^{**n-1}, c_h^{n+1}),$$

where $\overline{K}_{12} = \max(2K_1, K_2)$. To bound $\left\| \overline{\Delta}_2 c_h^{n+1} \right\|^2_{L^2(D)}$ in this expression, we note that

$$\left\| \overline{\Delta}_2 c_h^{n+1} \right\|^2_{L^2(D)} \leq 2 \left\| \overline{\Delta}_1 c_h^{n+1} \right\|^2_{L^2(D)} + 2 \left\| c_h^{*n} - c_h^{**n-1} \right\|^2_{L^2(D)},$$

making the change $y = X^{n,n+1}(x)$ and recalling Remark 11, we have that

$$\left\| c_h^{*n} - c_h^{**n-1} \right\|^2_{L^2(D)} = \int_D \left| c_h^n(X^{n,n+1}(x)) - c_h^{n-1}(X^{n-1,n+1}(x)) \right|^2 dx =$$
$$\int_D \left| c_h^n(y) - c_h^{n-1}(X^{n-1,n}(y) \right|^2 \left| J^{n,n+1} \right|^{-1} dy \leq \text{ (by 5.15)}$$
$$(1 + K_2\Delta t) \left\| c_h^n - c_h^{*n-1} \right\|^2_{L^2(D)} \leq$$
$$\left\| \overline{\Delta}_1 c_h^n \right\|^2_{L^2(D)} + \overline{K}_2 \Delta t \left(\|c_h^n\|^2_{L^2(D)} + \left\| c_h^{n-1} \right\|^2_{L^2(D)} \right),$$

where $\overline{K}_2 = K_2(1 + K_2\Delta t)$; hence

$$\left\| \overline{\Delta}_2 c_h^{n+1} \right\|^2_{L^2(D)} \leq 2 \left(\left\| \overline{\Delta}_1 c_h^{n+1} \right\|^2_{L^2(D)} + \left\| \overline{\Delta}_1 c_h^n \right\|^2 \right) + 2\overline{K}_2 \Delta t \left(\|c_h^n\|^2_{L^2(D)} + \left\| c_h^{n-1} \right\|^2_{L^2(D)} \right).$$

Therefore, we have:

$$\left\| \overline{\Delta}_1 c_h^{n+1} \right\|^2_{L^2(D)} - \frac{1}{4} \left\| \overline{\Delta}_2 c_h^{n+1} \right\|^2_{L^2(D)} \geq \frac{1}{2} \left(\left\| \overline{\Delta}_1 c_h^{n+1} \right\|^2_{L^2(D)} - \left\| \overline{\Delta}_1 c_h^n \right\|^2 \right)$$
$$-\frac{1}{2}\overline{K}_2 \Delta t \left(\|c_h^n\|^2 + \left\| c_h^{n-1} \right\|^2 \right). \quad (10.4)$$

It remains to bound the term $2(\widetilde{c}_h^{*n} - c_h^{*n}, c_h^{n+1}) - \dfrac{1}{2}(\widetilde{c}_h^{**n-1} - c_h^{**n-1}, c_h^{n+1})$ in (10.3). By virtue of Lemma 6 and using the inverse inequality (5.13), we have that

$$\begin{aligned}&\left|2(\widetilde{c}_h^{*n}-c_h^{*n},c_h^{n+1})-\frac{1}{2}(\widetilde{c}_h^{**n-1}-c_h^{**n-1},c_h^{n+1})\right|\leq\\&\left(2\left\|\widetilde{c}_h^{*n}-c_h^{*n}\right\|_{L^2(D)}+\frac{1}{2}\left\|\widetilde{c}_h^{**n-1}-c_h^{**n-1}\right\|_{L^2(D)}\right)\left\|c_h^{n+1}\right\|_{L^2(D)}\leq\\&2C_2h\Delta t(\left\|c_h^n\right\|_{L^2(D)}+\left\|c_h^{n-1}\right\|_{L^2(D)})\left\|c_h^{n+1}\right\|_{L^2(D)}\leq\\&2C_2h\Delta t\left(\left\|c_h^n\right\|_{L^2(D)}^2+\left\|c_h^{n-1}\right\|_{L^2(D)}^2\right)+C_2h\Delta t\left\|c_h^{n+1}\right\|_{L^2(D)}^2.\end{aligned}\tag{10.5}$$

Substituting (10.4) and (10.5) in (10.3) and summing from $n=1$ to $N-1$, we obtain

$$\begin{aligned}\sum_{n=1}^{N-1}\left(\Delta_1\left\|c_h^{n+1}\right\|_{L^2(D)}^2-\frac{1}{4}\Delta_2\left\|c_h^{n+1}\right\|_{L^2(D)}^2\right)=\frac{3}{4}\left\|c_h^N\right\|_{L^2(D)}^2-\frac{1}{4}\left\|c_h^{N-1}\right\|_{L^2(D)}^2-\\\frac{3}{4}\left\|c_h^1\right\|_{L^2(D)}^2+\frac{1}{4}\left\|c_h^0\right\|_{L^2(D)}^2,\end{aligned}$$

and

$$\begin{aligned}&\sum_{n=1}^{N-1}\left(\left\|\overline{\Delta}_1c_h^{n+1}\right\|_{L^2(D)}^2-\frac{1}{4}\left\|\overline{\Delta}_2c_h^{n+1}\right\|_{L^2(D)}^2\right)\geq\\&\frac{1}{2}\sum_{n=1}^{N-1}\left(\left\|\overline{\Delta}_1c_h^{n+1}\right\|_{L^2(D)}^2-\left\|\overline{\Delta}_1c_h^n\right\|_{L^2(D)}^2\right)-\frac{1}{2}\overline{K}_2\Delta t\sum_{n=1}^{N-1}\left(\left\|c_h^n\right\|_{L^2(D)}^2+\left\|c_h^{n-1}\right\|_{L^2(D)}^2\right)=\\&\frac{1}{2}\left(\left\|\overline{\Delta}_1c_h^N\right\|_{L^2(D)}^2-\left\|\overline{\Delta}_1c_h^1\right\|_{L^2(D)}^2\right)-\frac{1}{2}\overline{K}_2\Delta t\sum_{n=1}^{N-1}\left(\left\|c_h^n\right\|_{L^2(D)}^2+\left\|c_h^{n-1}\right\|_{L^2(D)}^2\right).\end{aligned}$$

Hence, we have:

$$\begin{aligned}&\left\|c_h^N\right\|_{L^2(D)}^2+\frac{2}{3}\left\|\overline{\Delta}_1c_h^N\right\|_{L^2(D)}^2+\frac{1}{4}\left\|c_h^0\right\|_{L^2(D)}^2+\Delta t\frac{4\nu_0^{1/2}}{3}\sum_{n=1}^{N-1}\left\|\nabla c_h^{n+1}\right\|_{L^2(D)}^2\leq\frac{1}{3}\left\|c_h^{N-1}\right\|^2+\\&\frac{1}{3}\left\|c_h^1\right\|_{L^2(D)}^2+\frac{1}{3}\left\|c_h^0\right\|_{L^2(D)}^2+\frac{4\Delta t}{3}\sum_{n=1}^{N-1}\left\|f^{n+1}\right\|_{L^2(D)}\left\|c_h^{n+1}\right\|_{L^2(D)}+\overline{C}\Delta t\sum_{n=0}^{N-1}\left\|c_h^{n+1}\right\|_{L^2(D)}^2,\end{aligned}$$

where $\overline{C}=8/3\overline{K}_{12}+4/3\overline{K}_2+7C_2h$. Let m be such that $\|c_h^m\|_{L^2(D)}=\max_{0\leq n\leq N}\|c_h^n\|_{L^2(D)}$, then

$$\begin{aligned}&\left\|c_h^m\right\|_{L^2(D)}^2\leq\frac{1}{3}\left(\left\|c_h^m\right\|_{L^2(D)}^2+\left\|c_h^1\right\|_{L^2(D)}^2+\left\|c_h^0\right\|_{L^2(D)}^2\right)+\\&\frac{4\Delta t}{3}\sum_{n=1}^{N-1}\left\|f^{n+1}\right\|_{L^2(D)}\left\|c_h^m\right\|_{L^2(D)}+C\Delta t\sum_{n=0}^{N-1}\left\|c_h^{n+1}\right\|_{L^2(D)}\left\|c_h^m\right\|_{L^2(D)},\end{aligned}$$

whence

$$\left\|c_h^N\right\|_{L^2(D)} \leq \frac{1}{2}\left(\|c_h^1\|_{L^2(D)} + \|c_h^0\|_{L^2(D)}\right) +$$

$$2\Delta t \sum_{n=1}^{N-1} \|f^{n+1}\|_{L^2(D)} + \overline{C}\Delta t \sum_{n=0}^{N-1} \left\|c_h^{n+1}\right\|_{L^2(D)},$$

since $\left\|c_h^N\right\|_{L^2(D)} \leq \|c_h^m\|_{L^2(D)}$. Applying Gronwall inequality the result (10.1) follows with $K = exp(\overline{C}T)$. □

Next, we estimate the error of the method in the L^2-norm. To do so, we set for each n

$$c^{n+1} - c_h^{n+1} = \left(c^{n+1} - R_h c^{n+1}\right) + (R_h c^{n+1} - c_h^{n+1}) := \rho^{n+1} + \theta_h^{n+1}. \tag{10.6}$$

As we see below, we need estimates of the term $\rho^n - \rho^n(X^{n,n+1}(x))$ which are established in the following lemma that is an extension of Lemma 1 in [14].

Lemma 11. *For all n, $\rho^n - \rho^n(X^{n,n+1}(x))$ satisfies the following bounds:*

(A)

$$\left\|\rho^n - \rho^n(X^{n,n+1}(x))\right\|_{H^{-1}(D)} \leq K_4 \Delta t \|\rho^n\|_{L^2(D)}, \tag{10.7a}$$

(B)

$$\left\|\rho^n - \rho^n(X^{n,n+1}(x))\right\|_{L^2(D)} \leq K_5 \Delta t \|\nabla \rho^n\|_{L^2(D)}, \tag{10.7b}$$

(C)

$$\left\|\rho^n - \rho^n(X^{n,n+1}(x))\right\|_{L^2(D)} \leq K_6 \|\rho^n\|_{L^2(D)}, \tag{10.7c}$$

where

$$\begin{cases} K_4 = \|u\|_{L^\infty(0,T;L^\infty(D)^d))} + C_p K_2(1 + K_2\Delta t), \\ K_5 = (1 + K_2\Delta t)\|u\|_{L^\infty(0,T;L^\infty(D)^d))}, \\ K_6 = 1 + (1 + K_2\Delta t) \end{cases} \tag{10.8}$$

and $H^{-1}(D)$ denotes the dual space of $H_0^1(D)$.

Proof: (A) For completeness of these notes we shall write the proof (10.7a), which is inspired in the proof of Lemma 1 in [14]. Setting $y = X^{n,n+1}(x)$ and denoting by $\overline{X}^{n,n+1}(\cdot)$ the inverse of $X^{n,n+1}(\cdot)$, i.e., $\overline{X}^{n,n+1} \circ X^{n,n+1}(x) = x$,

$$\left\|\rho^n - \rho^n(X^{n,n+1}(x))\right\|_{H^{-1}(D)} = \sup_{\phi \in H_0^1(D)} \left(\|\phi\|_{H_0^1(D)}^{-1} \int_D \left(\rho^n(x) - \rho^n(X^{n,n+1}(x))\right)\phi(x)dx\right), \tag{10.9}$$

changing variables and taking into account that $\left(J^{n,n+1}\right)^{-1} \leq 1 + K_2\Delta t$ we have that

$$\int_D \left(\rho^n(x) - \rho^n(X^{n,n+1}(x))\right)\phi(x)dx \leq \int_D \rho^n(x)\phi(x)dx -$$

$$\int_D \rho^n(y)\phi(\overline{X}^{n,n+1}(y))dy - K_2\Delta t \int_D \rho^n(y)\phi(\overline{X}^{n,n+1}(y))dy \leq$$

$$\int_D \rho^n(y)\left(\phi(y) - \phi(\overline{X}^{n,n+1}(y))\right)dy - K_2\Delta t \int_D \rho^n(y)\phi(\overline{X}^{n,n+1}(y))dy.$$

To proceed further we need to estimate $\phi(y) - \phi(\overline{X}^{n,n+1}(y))$. In doing so, we note that for $0 < \alpha < 1$, $y(\alpha) = \alpha y - (1-\alpha)\overline{X}^{n,n+1}(y)$ defines a quasi-isometric homeomorphism from D into D (see [28]) and

$$\phi(y) - \phi(\overline{X}^{n,n+1}(y)) = (y - \overline{X}^{n,n+1}(y)) \cdot \int_0^1 \nabla\phi(y(\alpha))d\alpha,$$

moreover, by the definition of $X^{n,n+1}(x)$ it follows that

$$y - \overline{X}^{n,n+1}(y) = -\int_{t_n}^{t_{n+1}} \mathbf{u}(X(x, t_{n+1}; t), t)dt;$$

hence

$$\left\|\phi(y) - \phi(\overline{X}^{n,n+1}(y)\right\|_{L^2(D)} \leq \Delta t \left\|\mathbf{u}\right\|_{L^\infty(0,T;L^\infty(D)^2)} \left\|\nabla\phi\right\|_{L^2(D)}.$$

With these considerations in mind it follows that

$$\left|\int_D \rho^n(y)\left(\phi(y) - \phi(\overline{X}^{n,n+1}(y))\right)dy\right| \leq \Delta t \left\|\mathbf{u}\right\|_{L^\infty(0,T;L^\infty(D)^2)} \left\|\rho^n\right\|_{L^2(D)} \left\|\phi\right\|_{H_0^1(D)}.$$

On the other hand, we have that

$$\left\|\phi(\overline{X}^{n,n+1}(y)\right\|^2_{L^2(D)} = \int_D \left|\phi(\overline{X}^{n,n+1}(y)\right|^2 dy = \int_D \left|\phi(x)\right|^2 J^{n,n+1}dx \leq$$

$$(1 + K_2\Delta t)\textstyle\int_D \left|\phi(x)\right|^2 dx = (1 + K_2\Delta t)\left\|\phi\right\|^2_{L^2(D)},$$

so that, denoting by C_p the constant of the Poincaré inequality,

$$K_2\Delta t \left|\textstyle\int_D \rho^n(y)\phi(\overline{X}^{n,n+1}(y))dy\right| \leq K_2\Delta t \left\|\rho^n\right\|_{L^2(D)} \left\|\phi(\overline{X}^{n,n+1}(y)\right\|_{L^2(D)} \leq$$

$$\Delta t C_p K_2(1 + K_2\Delta t)\left\|\rho^n\right\|_{L^2(D)} \left\|\phi\right\|_{H_0^1(D)}.$$

Collecting these bounds in (10.9) yields

$$\left\|\rho^n - \rho^n(X^{n,n+1}(x))\right\|_{H^{-1}(D)} \leq \Delta t K_4 \left\|\rho^n\right\|_{L^2(D)},$$

where

$$K_4 = \left\|u\right\|_{L^\infty(0,T;L^\infty(D)^d)} + C_p K_2(1 + K_2\Delta t).$$

(B) Now, we recast $\rho^n - \rho^n(X^{n,n+1}(x))$ as

$$\rho^n(x) - \rho^n(X^{n,n+1}(x)) = \int_{t_n}^{t_{n+1}} \frac{d\rho(X(x, t_{n+1}; t), t_n)}{dt}dt,$$

and by Cauchy-Schwarz inequality we get

$$\left|\rho^n - \rho^n(X^{n,n+1}(x))\right|^2 \leq \Delta t \int_{t_n}^{t_{n+1}} |\mathbf{u}(X(x,t_{n+1};t),t) \cdot \nabla\rho(X(x,t_{n+1};t),t_n)|^2 \, dt,$$

so that

$$\|\rho^n - \rho^n(X^{n,n+1}(x))\|^2_{L^2(D)} \leq \Delta t \int_D \int_{t_n}^{t_{n+1}} |\mathbf{u}(X(x,t_{n+1};t),t) \cdot \nabla\rho(X(x,t_{n+1};t),t_n)|^2 \, dtdx \leq$$

$$\Delta t \, \|\mathbf{u}\|^2_{L^\infty(0,T;D)^d} \int_{t_n}^{t_{n+1}} \int_D |\nabla\rho(X(x,t_{n+1};t),t_n)|^2 \, dxdt \leq \text{ (setting } y = X(x,t_{n+1};t))$$

$$\Delta t \, \|\mathbf{u}\|^2_{L^\infty((0,T)\times D)^d} \int_{t_n}^{t_{n+1}} \int_D |\nabla\rho(y,t_n)|^2 \left(J^{t,n+1}\right)^{-1} dydt \leq \quad \text{(by virtue of (5.15))}$$

$$(1 + K_2\Delta t)\Delta t^2 \, \|u\|^2_{L^\infty((0,T)\times D)^d} \, \|\nabla\rho^n\|^2_{L^2(D)} \, .$$

Then, setting

$$K_5 = (1 + K_2\Delta t) \, \|u\|_{L^\infty(0,T;L^\infty(D)^d)}$$

the result follows.

(C) Now, we estimate $\|\rho^n - \rho^n(X^{n,n+1}(x))\|_{L^2(D)}$ as

$$\|\rho^n - \rho^n(X^{n,n+1}(x))\|_{L^2(D)} \leq \|\rho^n\|_{L^2(D)} + \|\rho^n(X^{n,n+1}(x))\|_{L^2(D)} \leq$$

$$\text{(by (5.15))} \; (2 + K_2\Delta t) \, \|\rho^n\|_{L^2(D)} = K_6 \, \|\rho^n\|_{L^2(D)} \, ,$$

where

$$K_6 = 2 + K_2\Delta t.$$

□

We are now in a condition to establish the theorem on the convergence of modified LG methods.

Theorem 12. *Assuming $c(x,t)$ is sufficiently smooth and $u \in (0,T;W^{2,\infty}(D)^d)$, then for the solution c_h^n calculated by (9.2a) it holds that*

$$\|c - c_h\|_{l^\infty(0,T;L^2(D))} \leq \left(C + F\frac{T^{1/2}}{\Delta t}\min\left(\overline{D}_1, \overline{D}_2\Delta t, \overline{D}_3\Delta t\right)\right) B_1 h^{m+1} +$$

$$F\left(B_2 h^{m+1} + T^{1/2}h^2 B_1 + \Delta t^2\left(\left\|\frac{D^3 c}{Dt^3}\right\|_{L^2(0,T;L^2(D))} + \left\|\frac{D^2 c}{Dt^2}\right\|_{L^2(0,t_1;L^2(D))}\right)\right), \tag{10.10}$$

where $B_1 = \|c\|_{L^\infty(0,T;H^{m+1}(D))}$, $B_2 = \|c\|_{L^2(0,T;H^{m+1}(D))} + \|c_t\|_{L^2(0,T;H^{m+1}(D))}$, $F = 3(1 + 2K_2\Delta t)\varepsilon_M \exp(BT)\max(1.5C, 2\sqrt{2}(1 + Ch^m), 1)$ *with* $B = \varepsilon_1 + \dfrac{\nu_0}{2L_c^2} + 10C_2h + 2K_2(1 + K_2\Delta t) + \max(2K_1.K_2)$, ε_1 *is a positive constant independent of* ν_0 *and*

$$\varepsilon_M = \max.\left(\varepsilon_1^{-1/2}, \sqrt{\frac{5}{2}}\left(\varepsilon_1 + \frac{\nu_0}{2L_c^2}\right)^{-1/2}\right),$$

$$\overline{D}_1 = \frac{5}{3}\left(\frac{1 + \frac{5}{3}K_2\Delta t}{1 + 2K_2\Delta t}\right), \quad \overline{D}_2 = \frac{\|u\|_{L^\infty(D\times(0,T))^d}}{h}$$

and

$$\overline{D}_3 = \varepsilon_1^{1/2}\sqrt{\frac{1}{\nu_0}\frac{\|u\|_{L^\infty(D\times(0,T))^d}}{(1+2K_2\Delta t)}}\left(1 + \frac{C_pK_2(1+K_2\Delta t)}{\|u\|_{L^\infty(D\times(0,T))^d}}\right).$$

Proof: Noticing that ρ^{n+1} is bounded by (8.4a), we have to estimate θ_h^{n+1}. To do so, we decompose the first term of (9.2a) as we did above in the analysis of stability and subtract from the resulting expression the equation for the elliptic projection of $c(x, t_{n+1})$ onto V_h,

$$a^{n+1}(R_hc^{n+1}, v_h) = a^{n+1}(c^{n+1}, v_h) =$$

$$(f^{n+1}, v_h) - \left(\frac{Dc}{Dt}\,|_{t_{n+1}}, v_h\right), \quad \text{for all } v_h \in V_h,$$

to obtain for $n = 1, 2, \ldots$

$$\frac{1}{2}\left(3\theta_h^{n+1} - 4\theta_h^{*n} + \theta_h^{**n-1}, v_h\right) + \Delta t a^{n+1}(\theta_h^{n+1}, v_h) = (w^{n+1}, v_h), \tag{10.11}$$

where $w^{n+1} = \sum_{i=1}^{6} w_i^{n+1}$, and

$$w_1^{n+1} = \Delta t\left(\frac{3c^{n+1} - 4c^{*n} + c^{**n-1}}{2\Delta t} - \frac{Dc}{Dt}\,|_{t=t_{n+1}}\right),$$

$$w_2^{n+1} = -\frac{1}{2}\left(3\rho^{n+1} - 4\rho^n + \rho^{n-1}\right),$$

$$w_3^{n+1} = 2\left(\widetilde{\rho}^{*n} - \rho^{*n}\right) + 2\left(c^{*n} - \widetilde{c}^{*n}\right),$$

$$w_4^{n+1} = -\frac{1}{2}\left(\widetilde{\rho}^{**n-1} - \rho^{**n-1}\right) - \frac{1}{2}\left(c^{**n-1} - \widetilde{c}^{**n-1}\right),$$

$$w_5^{n+1} = -2\left(\rho^n - \rho^{*n}\right) + \frac{1}{2}\left(\rho^{n-1} - \rho^{**n-1}\right),$$

and

$$w_6^{n+1} = -2(\theta_h^{*n} - \widetilde{\theta}_h^{*n}) + \frac{1}{2}(\theta_h^{**n-1} - \widetilde{\theta}_h^{**n-1}).$$

Setting $v_h = \theta_h^{n+1}$ in (10.11) and operating in the same way as in the stability proof, taking $\left\|\theta_h^0\right\|_{L^2(D)} = 0$, yields,

$$\begin{aligned}
&\frac{3}{4}\left\|\theta_h^N\right\|_{L^2(D)}^2 + \Delta t\nu_0 \textstyle\sum_{n=1}^{N-1}\left\|\nabla\theta_h^{n+1}\right\|_{L^2(D)}^2 \leq \frac{1}{4}\left\|\theta_h^{N-1}\right\|_{L^2(D)}^2 + \\
&\frac{3}{4}\left\|\theta_h^1\right\|_{L^2(D)}^2 + \frac{1}{2}\left\|\overline{\Delta}_1\theta_h^1\right\|_{L^2(D)}^2 + \textstyle\sum_{n=1}^{N-1}\left|\left(\sum_{i=1}^5(w_i^{n+1},\theta_h^{n+1})\right)\right| + \\
&\textstyle\sum_{n=1}^{N-1}\left|\left(4(\theta_h^{*n} - \widetilde{\theta}_h^{*n}) - (\theta_h^{**n-1} - \widetilde{\theta}_h^{**n-1}), \theta_h^{n+1}\right)\right| \\
&+ \left(2\overline{K}_{12} + \overline{K}_2\right)\Delta t \textstyle\sum_{n=1}^{N-1}\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2 .
\end{aligned} \tag{10.12}$$

Next, we have to estimate the terms on the right hand side and then use the same argument as we did at the end of the stability proof. Thus, applying a Taylor expansion with integral remainder along the characteristic curves we have that

$$\left\|w_1^{n+1}\right\|_{L^2(D)} \leq \sqrt{\frac{2}{5}}\Delta t^{5/2}\left\|\frac{D^3c}{Dt^3}\right\|_{L^2(t_{n-1},t_{n+1};L^2(D))},$$

then using the Cauchy-Schwarz and elementary inequalities we obtain

$$\left|(w_1^{n+1},\theta_h^{n+1})\right| \leq \frac{\Delta t^4}{5\varepsilon}\left\|\frac{D^3c}{Dt^3}\right\|_{L^2(t_{n-1},t_{n+1};L^2(D))}^2 + \frac{\varepsilon\Delta t}{2}\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2 . \tag{10.13}$$

To bound the term $\left|(w_2^{n+1},\theta_h^{n+1})\right|$ we note that by virtue of Cauchy-Schwarz inequality

$$\left|(w_2^{n+1},\theta_h^{n+1})\right| \leq \left\|w_2^{n+1}\right\|_{L^2(D)}\left\|\theta_h^{n+1}\right\|_{L^2(D)},$$

then using the elementary inequality we have that

$$\left\|w_2^{n+1}\right\|_{L^2(D)}\left\|\theta_h^{n+1}\right\|_{L^2(D)} \leq \frac{\Delta t}{8\varepsilon}\left\|3\frac{\rho^{n+1}-\rho^n}{\Delta t} - \frac{\rho^n-\rho^{n-1}}{\Delta t}\right\|_{L^2D)}^2 + \frac{\Delta t\varepsilon}{2}\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2 .$$

Noting that

$$\left\|3\frac{\rho^{n+1}-\rho^n}{\Delta t}\right\|_{L^2D)}^2 = \frac{9}{\Delta t^2}\int_D\left|\int_{t_n}^{t_{n+1}}\rho_t dt\right|^2 dx \leq \frac{9}{\Delta t}\int_{t_n}^{t_{n+1}}\left\|\rho_t\right\|_{L^2(D)}^2 dt$$

and similarly for $\left\|\frac{\rho^n-\rho^{n-1}}{\Delta t}\right\|_{L^2D)}^2$, we finally have the inequality

$$\left|(w_2^{n+1},\theta_h^{n+1})\right| \leq \frac{18}{8\varepsilon}\left\|\rho_t\right\|_{L^2(t_{n-1},t_{n+1};L^2(D))}^2 + \frac{\varepsilon}{2}\Delta t\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2 . \tag{10.14}$$

To estimate $\left|(w_3^{n+1},\theta_h^{n+1})\right|$ and $\left|(w_4^{n+1},\theta_h^{n+1})\right|$ we use conveniently Lemma 6 and then it follows that

$$\begin{aligned}&\left|(w_3^{n+1},\theta_h^{n+1})\right|+\left|(w_4^{n+1},\theta_h^{n+1})\right|\leq\varepsilon\Delta t\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2+\\&\frac{4C_2^2\Delta t}{\varepsilon}h^4\left(\|\nabla c\|_{L^\infty(0,T;L^2(D))}+\|\nabla\rho\|_{L^\infty(0,T;L^2(D))}\right)^2.\end{aligned}\tag{10.15}$$

By Lemma 6 we find that

$$\begin{aligned}&\left|\left(-4(\theta_h^{*n}-\widetilde{\theta}_h^{*n})+\theta_h^{**n-1}-\widetilde{\theta}_h^{**n-1},\theta_h^{n+1}\right)\right|\leq\\&4\Delta tC_2h\left(\|\theta_h^n\|_{L^2(D)}^2+\left\|\theta_h^{n-1}\right\|_{L^2(D)}^2\right)+2\Delta tC_2h\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2.\end{aligned}\tag{10.16}$$

To estimate $\left|(w_5^{n+1},\theta_h^{n+1})\right|$ we make use of Lema 11 and have the following estimates:
(A)

$$\begin{aligned}&\left|(w_5^{n+1},\theta_h^{n+1})\right|\leq3\Delta tK_4\left(\|\rho\|_{L^\infty(0,T;L^2(D))}\right)\left\|\theta_h^{n+1}\right\|_{H_0^1(D)}\leq\\&\frac{9K_4^2}{2\nu_0}\Delta t\,\|\rho\|_{L^\infty(0,T;L^2(D))}^2+\frac{\nu_0}{2}\Delta t\left\|\nabla\theta_h^{n+1}\right\|_{L^2(D)}^2+\frac{\nu_0}{2L_c^2}\Delta t\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2.\end{aligned}\tag{10.17a}$$

(B)

$$\begin{aligned}&\left|(w_5^{n+1},\theta_h^{n+1})\right|\leq K_7\Delta t\,\|\nabla\rho\|_{L^\infty(0,T;L^2(D))}\left\|\theta_h^{n+1}\right\|_{L^2(D)}\leq\\&\frac{K_7^2\Delta t}{2\varepsilon}\|\nabla\rho\|_{L^\infty(0,T;L^2(D))}^2+\frac{\varepsilon\Delta t}{2}\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2,\end{aligned}\tag{10.17b}$$

where $K_7=3\|u\|_{L^\infty(D\times(0,T))^d}(1+2K_2\Delta t)$.
(C)

$$\begin{aligned}&\left|(w_5^{n+1},\theta_h^{n+1})\right|\leq K_8\,\|\rho\|_{L^\infty(0,T;L^2(D))}\left\|\theta_h^{n+1}\right\|_{L^2(D)}\leq\\&\frac{K_8^2\Delta t}{2\varepsilon}\left\|\frac{\rho}{\Delta t}\right\|_{L^\infty(0,T;L^2(D))}^2+\frac{\varepsilon\Delta t}{2}\left\|\theta_h^{n+1}\right\|_{L^2(D)}^2,\end{aligned}\tag{10.17c}$$

where $K_8=5(1+\frac{3}{5}K_2\Delta t)$.

Substituting the estimates (10.13)-(10.16) and (10.17a) into (10.12), summing from

$n = 0$ to $N-1$, and setting $T = N\Delta t$ and $\varepsilon_1 = \varepsilon$ we have the following bound:

$$\begin{aligned} \frac{3}{4}\left\|\theta_h^N\right\|_{L^2(D)}^2 \leq \frac{1}{4}\left\|\theta_h^{N-1}\right\|_{L^2(D)}^2 + \frac{3}{4}\left\|\theta_h^1\right\|_{L^2(D)}^2 + \frac{1}{2}\left\|\overline{\Delta}_1\theta_h^1\right\|_{L^2(D)}^2 + \\ \frac{\Delta t^4}{5\varepsilon_1}\left\|\frac{D^3c}{Dt^3}\right\|_{L^2(0,T;L^2(D))}^2 + \frac{9}{8\varepsilon_1}\left\|\rho_t\right\|_{L^2(0,T;L^2(D))}^2 + \\ \frac{4}{\varepsilon_1}C_2^2T\left(\left\|\nabla c\right\|_{L^\infty(0,T;L^2(D)^d)} + \left\|\nabla\rho\right\|_{L^\infty(0,T;L^2(D)^d)}\right)^2 h^4 + \\ \frac{9}{2\nu_0}K_4^2\left\|\rho\right\|_{L^\infty(0,T;L^2(D))}^2 + \Delta t B\sum_{n=0}^N\left\|\theta_h^n\right\|_{L^2(D)}^2, \end{aligned} \tag{10.18}$$

where $B = \varepsilon_1 + \dfrac{\nu_0}{2L_c^2} + 10C_2h + 2\overline{K}_{12} + \overline{K}_2$ is a positive constant. Now, using the same argument as in the stability proof and applying Gronwall inequality it follows that:

(A)

$$\left\|\theta_h^N\right\|_{L^2(D)} \leq A_1 + \overline{A}_2 + \overline{A}_3 + \overline{A}_4 \exp BT) + \sqrt{\frac{9}{\nu_0}}T^{1/2}K_4Ch^{m+1}\left\|c\right\|_{L^\infty(0,T;H^{m+1}(D))}. \tag{10.19a}$$

If we use the bounds (10.17b) or (10.17c), respectively, instead of the bound (10.17a) we obtain

(B)

$$\left\|\theta_h^N\right\|_{L^2(D)} \leq A_1 + A_2 + A_3 + A_4 + \exp(BT)T^{1/2}K_7Ch^m\left\|c\right\|_{L^\infty(0,T;H^{m+1}(D))}. \tag{10.19b}$$

(C)

$$\left\|\theta_h^N\right\|_{L^2(D)} \leq A_1 + A_2 + A_3 + A_4 + \exp(BT)\frac{T^{1/2}}{\Delta t}K_8Ch^{m+1}\left\|c\right\|_{L^\infty(0,T;H^{m+1}(D))}, \tag{10.19c}$$

where the constant ε of the inequalities (10.13)-(10.15) is now, taking as ε another constant ε_2, such that $B = \varepsilon_1 + \dfrac{\nu_0}{2L_c^2} + 10C_2h + 2\overline{K}\ _{12} + \overline{K}_2 = \dfrac{5}{2}\varepsilon_2 + 10C_2h + 2\overline{K}\ _{12} + \overline{K}_2$, so that, $\varepsilon_2 = \dfrac{2}{5}\left(\varepsilon_1 + \dfrac{\nu_0}{2L_c^2}\right)$. Let $(F_1, F_2) = (\varepsilon_1^{-1/2}, \varepsilon_2^{-1/2})\exp(BT)$, the other constants are given by the formulae

$$A_1 = \exp(BT)\left(\frac{3}{2}\left\|\theta_h^1\right\|_{L^2(D)} + \left\|\overline{\Delta}_1\theta_h^1\right\|_{L^2(D)}\right),$$

$$(\overline{A}_2, A_2) = (F_1, F_2)\sqrt{\frac{2}{5}}\left\|\frac{D^3c}{Dt^3}\right\|_{L^2(0,T;L^2(D))}\Delta t^2,$$

$$(\overline{A}_3, A_3) = (F_1, F_2)\frac{3\sqrt{2}}{2}C\left(\|c\|_{L^2(0,T;H^{m+1}(D))} + \|c_t\|_{L^2(0,T;H^{m+1}(D))}\right)h^{m+1},$$

$$(\overline{A}_4, A_4) = (F_1, F_2)2\sqrt{2}C_2T^{1/2}\left(\|\nabla c\|_{L^\infty(0,T;L^2(D))} + Ch^m\|c\|_{L^\infty(0,T;H^{m+1}(D))}\right)h^2.$$

Hence, setting $\varepsilon_M = \max(\varepsilon_1^{-1/2}, \varepsilon_2^{-1/2})$ and considering that $\frac{3}{2}\left\|\theta_h^1\right\|_{L^2(D)} + \left\|\overline{\Delta}_1\theta_h^1\right\|_{L^2(D)} \leq \frac{5}{2}Ch^{m+1}\|c\|_{L^2(0,t_1;H^{m+1}(D))} + \Delta t^2\left\|\frac{D^2c}{Dt}\right\|_{L^2(0,t_1;L^2(D))}$, there exists a constant

$$F = 3(1+2K_2\Delta t)\varepsilon_M\exp(BT)\max(\frac{3\sqrt{2}C}{2},\ 2\sqrt{2}C_2(1+Ch^m),\ 1)$$

such that

$$\begin{aligned}\left\|\theta_h^N\right\|_{L^2(D)} \leq F\left(\|c\|_{L^2(0,T;H^{m+1}(D))} + \|c_t\|_{L^2(0,T;H^{m+1}(D))}\right)h^{m+1} + \\ F\Delta t^2\left(\left\|\frac{D^2c}{Dt^2}\right\|_{L^2(0,t_1;L^2(D))} + \left\|\frac{D^3c}{Dt^3}\right\|_{L^2(0,T;L^2(D))}\right) + FT^{1/2}\|c\|_{L^\infty(0,T;H^{m+1}(D))}h^2 + \\ F\frac{T^{1/2}}{\Delta t}\min\left(\overline{D}_1, \overline{D}_2\Delta t, \overline{D}_3\Delta t\right)h^{m+1}\|c\|_{L^\infty(0,T;H^{m+1}(D))},\end{aligned} \tag{10.20}$$

where

$$\overline{D}_1 = \frac{5}{3}\frac{1+\frac{5}{3}K_2\Delta t}{1+2K_2\Delta t}, \quad \overline{D}_2 = \frac{\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}{h}$$

and

$$\overline{D}_3 = \varepsilon_1^{1/2}\sqrt{\frac{1}{\nu_0}\frac{\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}{(1+2K_2\Delta t)}}\left(1 + \frac{C_pK_2(1+K_2\Delta t)}{\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}\right).$$

Then substituting (10.20) into (10.6) and using (8.4a) ends the proof. □

Remark 13. *Defining the local Péclet number as* $Pe = \frac{h\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}{2\nu_0}$, *we can write* $\overline{D}_3$ *as*

$$\overline{D}_3 = CPe^{1/2}\left(\frac{\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}{h}\right)^{1/2},$$

where the constant C *is given by*

$$C = \sqrt{2\varepsilon_1}\frac{1}{(1+2K_2\Delta t)}\left(1+\frac{C_pK_2(1+K_2\Delta t)}{\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}\right).$$

Thus, when $\dfrac{\overline{D}_3}{\overline{D}_2} > 1$*, or equivalently when the local Péclet number* $Pe > \dfrac{\|u\|_{L^\infty(D\times(0,T))^d}}{hC^2}$*, we have that the error is*

$$O(\max(h^m + h^2) + \Delta t^2),$$

when $\Delta t < \Delta t_C \simeq \dfrac{5h}{3\,\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}$*, whereas for* Δt *large, i.e.,* $\Delta t > \Delta t_C$ *the error is*

$$O(\max((\frac{h^{m+1}}{\Delta t} + h^2), \Delta t^2)).$$

In both cases the constants of the error do not depend on ν_0^{-1} *and therefore, when* $\nu_0 \to 0$ *the error tends to that of the pure convective problem. However, when* $\dfrac{\overline{D}_3}{\overline{D}_2} < 1$*, i.e.,*

$$Pe < \frac{\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}{hC^2} \quad \text{or} \quad 0 < \frac{C^2h^2}{2} < \nu_0,$$

we have that for Δt *sufficiently small, i.e.,* $\Delta t < \dfrac{\overline{D}_1}{\overline{D}_3} = \dfrac{1}{C}\left(\dfrac{h}{Pe\,\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}\right)^{1/2} = \Delta t_D$*, the error is*

$$O(\max((h^{m+1} + h^2), \Delta t^2)),$$

and the constants of the error are multiplied by $\nu_0^{-1/2}$*, which in general is not that large because this case corresponds to diffusion dominated problems. Finally, for* $\Delta t > \Delta t_D$*, the error is*

$$O(\max((\frac{h^{m+1}}{\Delta t} + h^2), \Delta t^2)).$$

As a final remark we should add the following.

Remark 14. *The term* $O(h^2)$ *which appears in Theorem* 12 *is specific of the modified LG method; thus, Theorem* 12 *without such a term is the estimate for the error of the conventional LG method, and therefore, the above remarks are also valid for this method.*

11 Numerical tests

We test the performance of the modified LG method as well as the validity of the theoretical convergence results of Theorem 12. To this end we consider the model problem (7.1) in the whole space $\mathbb{R}^3$ and $t \geq 0$, with data $f(x,t) = 0$, $\nu(x,t) = \nu_0$, velocity vector $u(x,t) = (-x_2, x_1, 0)$ and initial condition

$$c(x,0) = \exp\left\{-\frac{(x_1+0.25)^2 + {x_2}^2 + x_3^2}{\sigma_0^2}\right\}, \tag{11.1}$$

where $\sigma_0 = 0.08$. It is known that the analytical solution is given by

$$c(x,t) = \left(\frac{\sigma_0^2}{\sqrt{\sigma_0^2 + 2\nu_0 t}}\right)^3 \exp\left\{-\frac{(\overline{x}_1(t)+0.25)^2 + \overline{x}_2^2(t) + \overline{x}_3^2(t)}{\sigma_0^2}\right\} \tag{11.2}$$

where $\overline{x}_1(t) = x_1 \cos t + x_2 \sin t$, $\overline{x}_2(t) = -x_1 \sin t + x_2 \cos t$ and $\overline{x}_3(t) = x_3$. Our numerical test consists of solving the model problem (7.1), the analytical solution of which is (11.2), in the bounded region $\overline{D} := [-0.9, 0.9]^2 \times [-0.3, 0.3]$ and time interval $I := [0, 2\pi]$ with the same data as in the problem formulated in the whole of $\mathbb{R}^3$, but with boundary conditions given by (11.2) when $(x,t) \in \partial D \times [0, 2\pi]$. We calculate the numerical solution in a family of meshes $(D_h)_j$ formed by tetrahedra and P_2 conforming finite element spaces V_h with different time steps Δt_k. The values of h_j and Δt_k vary according to the relations $(h_j, \Delta t_k) = \frac{1}{\sqrt{2}}(h_{j-1}, \Delta t_{k-1})$, $j = 1, 2, \ldots$, and $k = 1, 2, \ldots$. The integrals are calculated by the Gauss-Legendre quadrature rules of order 6 presented in [25], these quadrature rules have positive weights. Table 7 shows the main features of the meshes employed in the numerical tests.

Mesh	Elements	Vertices	Nodes	h
1	2565	660	4302	0.169705627
2	6320	1463	9995	0.12
3	17623	3718	26528	0.084852814
4	51268	10106	74479	0.06
5	141530	26619	200647	0.042426407
6	415405	75320	578044	0.03

Table 7: Features of the meshes used in the experiments

The results we show below have been obtained by the modified LG-BDF2 method with the diffusion coefficients $\nu_0 = 10^{-2}$, 10^{-3}, 10^{-4}, and 10^{-5}; the solution of (2.2) taking as initial condition the vertices of the tetrahedra is calculated by a Runge-Kutta method of order 3 combined with the search-locate algorithm of [1]; so that the error committed in the calculation of the feet of the characteristics is negligible when compared with the error of the method. We must also say that we could calculate once and for all the feet of the characteristics because the velocity vector does not depend on time; however, when we compare the performances of the modified and conventional LG methods, see Figure 7, we calculate the solution of (2.2) every time step in order to compute the CPU time of each method. For the values of h used in the experiments the values of $\nu_0 = 10^{-4}$ and 10^{-5} yield high local Péclet numbers; so that, according to Theorem 12 the behavior of the error corresponds to the case $\frac{\overline{D}3}{\overline{D}_2} > 1$, except possibly for $\nu_0 = 10^{-3}$, and certainly for $\nu_0 = 10^{-2}$, these two values of diffusion coefficients yield low local Péclet numbers, and consequently, the error behavior for them corresponds to the case $\frac{\overline{D}3}{\overline{D}_2} < 1$. We show in Figures 4 and 5 the variation of the relative error, $\frac{\|c(x,t) - c_h(x,t)\|_{L^2(D)}}{\|c(x,t)\|_{L^2(D)}}$, as a function of Δt (or equivalently the number of time steps) with $h = 0.03$. We observe in Figure 4 that when Δt is large enough, i.e., $\Delta t > \Delta t_C$ the error is $O(max((\frac{h^3}{\Delta t} + h^2), \Delta t^2))$, however when $\Delta t < \Delta t_C$ the error remains almost constant as Δt decreases, this indicates that the error does not depend on Δt as Theorem 12 predicts, actually the error is $O(h^2)$.

Figure 5 shows the relative error behavior for $\nu_0 = 10^{-3}$ (upper) and $\nu_0 = 10^{-2}$ (lower). For such values of ν_0 the ratio $\frac{\overline{D}3}{\overline{D}_2} < 1$ and consequently we observe in the graphs that when Δt is sufficiently large, or equivalently, when the number of time steps is less than 10, the convergence is $O(max((\frac{h^{m+1}}{\Delta t} + h^2), \Delta t^2))$; however, when $0.1 \leq \Delta t$ the convergence is given by $O(max((h^{m+1} + h^2), \Delta t^2))$, this means that when $0.1 \leq \Delta t \leq h$ the error is $O(\Delta t^2)$, and when $\Delta t < h$, i.e., when the number of time steps is larger than, say, $\simeq 30$ the error is $O(h^2)$. We further observe in Figure 4 and the upper panel of Figure 5 that when Δt is so small that the CFL number is $\ll 1$ the error increases slowly. We attribute this increase to the possible existence of instabilities at $CFL < 1$ in the LG methods when the integrals $\int_D c_h^n(X^{n,n+1}(x))v_h(x)dx$ are approximated by Gaussian quadrature rules instead of being calculated exactly, see [22]. It occurs that if the diffusion is sufficiently large then it kills the instability, as happens when $\nu_0 = 10^{-2}$; but when the diffusion is sufficiently small the instability does not die out and will grow slowly, and eventually the calculations may blow up after many time steps. To ascertain that this phenomena may be present at low CFL numbers when the diffusion coefficient are very small, we repeat the numerical experiments with $\nu_0 = 10^{-4}$ using a Gauss-Legendre quadrature of order 8. We observe in Figure 6 that this quadrature rule delays the appearance of the instability and this is weaker than the one which appears when the integrals are calculated by the quadrature rule of order 6. Next, we compare in Figure 7 the conventional LG-BDF2 and the modified LG-BDF2 methods showing the variation of the relative error as a function of h for a time step $\Delta t = 0.063$. We see that both methods converge with the same order.

As for the CPU time, the modified LG-BDF2 method needs 20 minutes and 3 seconds, whereas the LG-BDF2 methods needs 56 minutes and 27 seconds, to complete 100 time steps in the mesh number 6, i.e., $h = 0.03$.

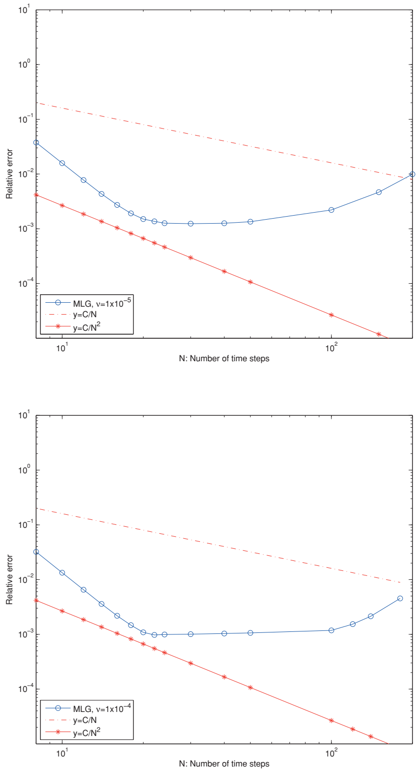

Figure 4: Variation of the relative error as a function of Δt for $h = 0.03$ and $\nu_0 = 1 \times 10^{-5}$ (upper) and $\nu_0 = 1 \times 10^{-4}$ (lower).

12 Conclusions

(1) The LG methods are powerful tools to solve convection dominated-diffusion equations if they are implemented properly, because they require the use of high order quadrature rules in order to be stable and accurate. One of the main points of these notes has been the presentation of algorithms to achieve an efficient implementation of LG methods. In this way, we have introduced a modified LG method that combined with conforming P_1 and P_2 finite elements is as accurate as the conventional LG methods, but it is much more efficient because it has to calculate less departure points. Numerical experiments in 3D problems show that the new methods might be 3 times faster than the conventional LG

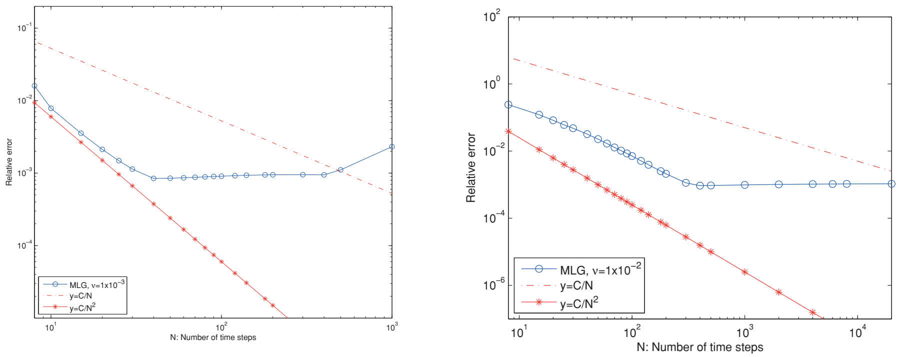

Figure 5: Variation of the relative error as a function of Δt for $h = 0.03$ and $\nu_0 = 1 \times 10^{-3}$ (upper) and $\nu_0 = 1 \times 10^{-2}$ (lower).

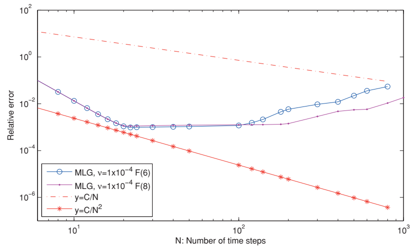

Figure 6: Comparison of the results obtained with Gauss-Legendre quadrature rules of order 8 and 6 when $\nu_0 = 10^{-4}$

methods.

(2) The error analysis of LG methods for convection-diffusion problems shows that the error constants are uniformly bounded in relation with the diffusion coefficient ν_0, this means that when this coefficient goes to zero the constants remains bounded and the error tends to the error of the pure convection problem.

(3) Our study reveals the following scenario: (a) when $\nu_0 < \dfrac{C^2h^2}{2}$ and $\Delta t < \Delta t_c = K\dfrac{h}{\|\mathbf{u}\|_{L^\infty(D\times(0,T))^d}}$, the convergence is of the form $O(max((h^m + h^2), \Delta t^q))$, $q = 1$ or 2 depending on the order of the method, here C is a bounded constant depending on $|D|\,\|\mathbf{u}\|^{-1}_{L^\infty(D\times(0,T))^d}$ and K is another constant close to 1; (b) when $\nu_0 < \dfrac{C^2h^2}{2}$ and

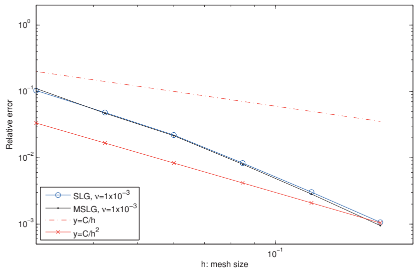

Figure 7: Variation of relative error of LG-BDF2(LG) and modfied LG-BDF2 (MLG) methods as a function of h for $\Delta t = 0.063$ an $\nu = 1 \times 10^{-3}$

$\Delta t > \Delta t_C$, that is when Δt is large enough, the error is $O(max((\frac{h^{m+1}}{\Delta t} + h^2), \Delta^q))$; (c) when $\nu_0 > \frac{C^2 h^2}{2}$ and $\Delta t < \Delta t_D = \frac{1}{C}\left(\frac{2\nu_0}{\|u\|^2_{L^\infty(D\times(0,T))^d}}\right)^{1/2}$ the convergence is now of the form $O(max((h^{m+1} + h^2), \Delta t^q))$; finally (d) when $\nu_0 > \frac{C^2 h^2}{2}$ and $\Delta t > \Delta t_D$ the method converges with as $O(max(\frac{h^{m+1}}{\Delta t}, h^2), \Delta^q))$. Since the term $O(h^2)$ is specific of the modified LG methods, then examining back the proofs of our error analysis we can conclude that the conventional LG methods have the same error behavior as the modified LG methods but without the term h^2. Consequently, we can say that our convergence result extends the convergence results of [14] and [23] establishing the regime of the validity of such results.

Acknowledgements

The author acknowledges the financial support by Ministerio de Ciencia e Innovación de España via the grant CGL2007-66440-CO4-01.

References

[1] A. Allievi and R. Bermejo, *A generalized particle search-locate algorithm for arbitrary grids*. J. Comp. Physics, **132** (1997), pp.157-166

[2] M. Bause and P. Knabner, *Uniform error analysis for Lagrange-Galerkin approximations of convection-dominated problems*. SIAM J. Numer. Anal., **39** (2002), 1954-1984.

[3] M. Benítez, *Métodos Numéricos para Problemas de Convección Difusión. Aplicación*

a la Convección Natural, PhD. Thesis. Departamento de Matemática Aplicada. Universidade de Santiago de Compostela (2009).

[4] J.P. Benque, G. Labadie and J. Ronat, *A new finite element method for the Navier-Stokes equations with a temperature equation.* Proccedings of the 4th International Symposium on Finite Element Methods in Flow Problems (T. Kawai, ed.). Amsterdam, The Netherlands: North-Holland (1982), 295-301.

[5] A. Bermúdez, M. R. Nogueiras and C. Vázquez, *Numerical analysis of convection-diffusion-reaction problems with higher order characteristics/finite elements. Part I: Time discretization.* SIAM J. Numer. Anal., **44** (2006) 1829-1853.

[6] A. Bermúdez, M. R. Nogueiras and C. Vázquez, *Numerical analysis of convection-diffusion-reaction problems with higher order characteristics/finite elements. Part II: Fully discretized schemes and quadratuire formulas*. SIAM J. Numer. Anal., **44** (2006) 1854-1876.

[7] R. Bermejo and L. Saavedra, *Modified Lagrange-Galerkin methods of first and second order in time for convection-diffusion problems.* Numer. Math. (Submitted).

[8] R. Bermejo and J. Carpio, *A semi-Lagrangian-Galerkin projection scheme for convection equations.* IMA Journal of Numerical Analysis. (2010) doi:101093/imamum/drn044

[9] R. Bermejo and J. Carpio, *An adaptive finite element semi-Lagrangian implicit-explicit Runge-Kutta-Chebyshev method for convection dominated reaction-diffusion problems*. Appl. Numer. Math., **58** (2008) 16-39.

[10] R. Bermejo, Analysis of a class of quasi-monotone and conservative semi-Lagrangian advection schemes. Numer. Math., **87** (2001), 597-623.

[11] C. Bernardi, *Optimal finite-element interpolation on curved domains*. SIAM J. Numer. Anal., **36** (1989), 1212-1240.

[12] K. Boukir, Y. Maday, B. Métivet and E. Razanfindrakoto, *A high-order characteristics/finite element method for the incompressible Navier-Stokes equations.* International J. Numer. Methods Fluids., **25** (1997), 1421-1454.

[13] A. N. Brooks and T. J. R. Hughes, *Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations.* Comput. Meth. Appl. Mech. Engrg. **32** (1982) 199-259.

[14] J. Douglas and T.F. Russell, *Numerical methods for convection-dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures.* SIAM J. Numer. Anal., **19** (1982), 871-885.

[15] R. Ewing and T.F. Russell, *Multistep Galerkin methods along characteristics for convection-diffusion problems,* in Advances in Computer Methods for Partial Differential Equations IV, R. Vchtneveski and R.S. Stepleman, eds., IMACS, New Brunswick, NJ. (1981), 28-36

[16] M. Falcone and R. Ferreti, *Convergence analysis for a clas of high-order semi-Lagrangian advection schemes.* SIAM J. Numer. Anal., **35** (1998), 909-940.

[17] P. Hansbo, *The characteristic streamline diffusion method for convection-diffusion problems*. Comput. Meth. Appl. Mech. Engrg. **96** (1992) 239-253.

[18] P. Hansbo, *The characteristic streamline diffusion method for time-dependent incompressible Navier-Stokes equations*. Comput. Meth. Appl. Mech. Engrg., **99** (1992) 171-186.

[19] T. J. R. Hughes, L.P. Franca and G.M. Hulbert, *A new finite element formulation for computational fluid dynamics. VIII. The Galerkin/least-squares method for advectio-diffusive equations*. Comput. Meth. Appl. Mech. Engrg., **73** (1989) 173-189.

[20] C. Johnson, *A new approach to algorithms for convection problems which are based on exact transport + projection.* Comput. Meth. Appl. Mech. Engrg. **100** (1992) 45-62.

[21] K.W. Morton, *Numerical Solution of Convection-Diffusion Problems*, Chapman and Hall, London, 1996.

[22] K. W. Morton, A. Priestley and E. Süli, *Stability of the Lagrange-Galerkin method with non-exact integration.* $M^2NA.$, **22** (1988), 625-653.

[23] O. Pironneau, *On the transport-diffusion algorithm and its applications to the Navier-Stokes equations*. Numer. Math., **38** (1982), 309-332.

[24] A. Quarteroni and A. Valli, *Numerical Approxiamtion of partial Differential Equations*, Springer-Verlag, Berlin Heidelberg (2008).

[25] H. T. Rathod, B. Venkatesudu, and K. V. Nagaraja, *Gauss-Legendre quadrature formulas over a tetahedron. Numerical Methods for Partial Differential Equations*. **22** (2006), 197-219.

[26] H. Rui and M. Tabata, *A second order characteristic finite element scheme for convection-diffusion problems*. Numer. Math., **92** (2002), 161-177.

[27] A. Staniforth and J. Cote, *Semi-Lagrangian integration schemes for the atmospheric models: A review.* Mon. Wea. Rev., **119** (1991), 2206-2223.

[28] E. Süli, *Convergence and nonlinear stability of the Lagrange-Galerkin method for the Navier-Stokes equations*. Numer. Math., **53** (1988), 459-483.