A very high-order finite volume method for the time-dependent convection-diffusion problem with Butcher tableau extension

S. Clain[a,b], G.J. Machado[a,*]

[a] *Centre of Mathematics, University of Minho, Campus de Azurém, Guimarães, Portugal*
[b] *Institut de Mathématiques de Toulouse, Université de Toulouse, Toulouse, France*

Abstract

The time discretization of a very high-order finite volume method may give rise to new numerical difficulties resulting into accuracy degradations. Indeed, for the simple one-dimensional unstationary convection-diffusion equation for instance, a conflicting situation between the source term time discretization and the boundary conditions may arise when using the standard Runge-Kutta method. We propose an alternative procedure by extending the *Butcher Tableau* to overcome this specific difficulty and achieve fourth-, sixth- or eighth-order of accuracy schemes in space and time. To this end, a new finite volume method is designed based on specific polynomial reconstructions for the space discretization, while we use the *Extended Butcher Tableau* to perform the time discretization. A large set of numerical tests has been carried out to validate the proposed method.

Keywords: finite volume, very high-order, convection-diffusion, polynomial reconstruction, Butcher Tableau.

1. Introduction

Very high-order finite volume schemes involving diffusive or viscous contributions is a recent but important challenge [1, 2] (see [3] for a generic framework of the finite volume method with diffusion terms). In [4], a new high-order finite volume method has been developed providing up to a sixth-order approximation for the two-dimensional steady-state convection-diffusion problem. The method is based on specific polynomial reconstructions used for the convection and the diffusion fluxes [5, 6, 7, 2, 8]. The issue we address here is the extension to the time-dependent case. The semi-discretization in space results in an initial value Ordinary Differential Equation (ODE) system where standard methods produce a fully discretized system. The Runge-Kutta method [9] is one of the most popular schemes but alternative techniques such as the strong stability preserving time discretization schemes [10] or the ADER method [11, 12, 8] may be considered. Nevertheless, a straightforward application of the time discretization of the

*Corresponding author: Phone: +351 253 510 445
Email addresses: clain@math.uminho.pt (S. Clain), gjm@math.uminho.pt (G.J. Machado)

ODE is not fully satisfactory since, as we shall prove in the paper, accuracy deterioration is observed when dealing with fourth-, sixth- or eighth-order methods. Indeed, numerical problems appear close to the boundary resulting from an inadequacy between the time discretization of the diffusion term of the differential equation and the Dirichlet or Neumann boundary conditions where such a phenomenon arises when dealing with time dependent boundary conditions. The goal of the present study is to overcome this problem introducing two time-discretizations (one for the differential equation and another for the boundary conditions) such that they match in the sense that a class of functions (for example constant in space) are exactly solved. We present the method for one-dimensional geometries for the sake of simplicity since the main difficulty concerns the time discretization, being the extension for higher spatial dimension the subject of a future work.

The paper is organized as follows. In the second section, we introduce the notations and ingredients to perform the space discretization for the steady-state problem rephrasing [4] for the one-dimensional context. The motivation is that the time discretization is based on the stationary case by using the method of lines. Section three is dedicated to the design of high-order finite volume schemes for the time-dependent case where we develop a new class of time discretization to preserve the compatibility between the source term and the boundary conditions. Section four deals with the numerical tests showing that the scheme provides fourth-, sixth- or eighth-order accuracy both in space and time. We propose examples which show that the straightforward time discretization may not provide the expected order whereas the new method effectively gives the optimal order of convergence or a significant improvement. In the last section we present the conclusions of the work.

2. Steady-state convection-diffusion problem

Very high-order finite volume discretization in space is introduced in the present section. We first deal with the steady-state context since it is an important building-block for the time-dependent problem we shall tackle in the next section. The main ingredient is a polynomial reconstruction procedure which provides accurate representations of the solution further used to evaluate the flux for the convection and the diffusion contributions.

2.1. Finite volume discretization

We seek function $\phi = \phi(x)$ solution of the steady-state convection-diffusion equation

$$(v\phi)' - (\kappa\phi')' = f, \quad \text{in } \Omega, \tag{1}$$

where $\Omega = (x_{\mathrm{L}}, x_{\mathrm{R}})$ is an open bounded interval of $\mathbb{R}$, the diffusive coefficient $\kappa = \kappa(x)$ and the convective coefficient $v = v(x)$ are regular functions on $\overline{\Omega}$ with $\kappa(x) \geqslant \kappa_0 > 0$, and $f = f(x)$ represents a regular source term. We shall consider three different types of boundary conditions:

- Dirichlet: $\phi(\bar{x}) = \phi_{\mathrm{D}}(\bar{x})$;
- total Neumann: $v(\bar{x})\phi(\bar{x}) - \kappa(\bar{x})\phi'(\bar{x}) = \phi_{\mathrm{T}}(\bar{x})$;

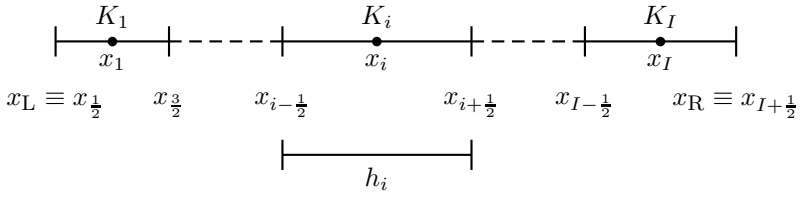

Figure 1: Notation for the mesh, cells, and interfaces.

- partial Neumann: $-\kappa(\bar{x})\phi'(\bar{x}) = \phi_{\mathrm{P}}(\bar{x})$,

where $\bar{x}$ is a boundary point and $\phi_{\mathrm{D}} = \phi_{\mathrm{D}}(x)$, $\phi_{\mathrm{T}} = \phi_{\mathrm{T}}(x)$, and $\phi_{\mathrm{P}} = \phi_{\mathrm{P}}(x)$ are given functions (only defined at points x_{L} and x_{R} for the one-dimensional case, being the extension for the two-dimensional situation straightforward). Other boundary conditions as the Robin condition can also be considered (see [13] for instance).

We underline the main difference between the total Neumann condition and the partial Neumann condition. In the first case, the natural flux involving both the convection and the diffusion are linked to a given function ϕ_{T}, whereas in the second case only the diffusive part is controlled by an external function ϕ_{P}. We shall see in the following that Neumann conditions shall bring numerical difficulties when dealing with large Péclet numbers.

To perform the space discretization we denote by $\mathcal{T}_h$ a mesh of the interval $\overline{\Omega}$ constituted of cells $K_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, $i = 1, \ldots, I$, with centroid x_i, where $x_{\frac{1}{2}} = x_{\mathrm{L}}$, $x_{I+\frac{1}{2}} = x_{\mathrm{R}}$, and $x_{i+\frac{1}{2}} = x_{i-\frac{1}{2}} + h_i$, $i = 1, \ldots, I$ stand for the interfaces (*cf.* Fig. 1).

Using the classical finite volume methodology, equation (1) is integrated over cell K_i, $i = 1, \ldots, I$, resulting in

$$\frac{1}{h_i}\left(\mathbb{F}_{i+\frac{1}{2}} - \mathbb{F}_{i-\frac{1}{2}}\right) - f_i = 0, \tag{2}$$

where we set the physical flux as

$$\mathbb{F}_{i\pm\frac{1}{2}} = v(x_{i\pm\frac{1}{2}})\phi(x_{i\pm\frac{1}{2}}) - \kappa(x_{i\pm\frac{1}{2}})\phi'(x_{i\pm\frac{1}{2}})$$

and the mean value of the source term as

$$\bar{f}_i = \frac{1}{h_i}\int_{K_i} f(\xi)\,\mathrm{d}\xi.$$

The exact mean source term $\bar{f}_i$ is approximated by f_i through Gaussian quadrature approximation and the numerical approximation for the exact fluxes $\mathbb{F}_{i\pm\frac{1}{2}}$ is the subject of the next two subsections.

2.2. Polynomial reconstructions

To achieve high-order numerical approximations, we introduce local polynomial reconstructions of the underlying solution whereas second-order popular techniques use local interpolation based on centroids or vertices (see [14] for the two-dimensional case).

We adapt the reconstruction procedure developed in [4] for the one-dimensional situation and highlight three specific and important points: interpolations are carried out with mean values such that the mass matrix deriving from time discretization turns out to be a diagonal one (see next section); conservative polynomials are well adapted for convection fluxes while non-conservative polynomials associated to the interfaces are required for the diffusive contribution; and the weights distribution in the minimization process is crucial to provide a positivity preserving scheme when dealing with pure diffusive problems.

At the first stage, we define the stencils associated to the cells and the interfaces. For any cell K_i, $i = 1, \ldots, I$, and any degree d of the polynomial reconstruction, we shall denote by S_i the stencil composed of the even n closest neighbor cells (excluding cell K_i). In the same way, for any interface $x_{i+\frac{1}{2}}$, $i = 0, \ldots, I$, and and any degree d of the polynomial reconstruction, we denote by $S_{i+\frac{1}{2}}$ the stencil constituted of the even n closest neighbor cells of the interface.

The second stage consists in defining the polynomial reconstructions based on the data of the associated stencil. To this end, we assume that $\Phi = (\phi_i)_{i=1,\ldots,I}$, $\phi_i \in \mathbb{R}$, is a vector gathering approximations of the mean value of ϕ over cells K_i, $i = 1, \ldots, I$. We have to consider three cases:

- Conservative polynomial reconstruction on cells: let $i \in \{1, \ldots, I\}$. The polynomial reconstruction of degree d associated to cell K_i is defined as

$$\boldsymbol{\phi}_i(x) = \phi_i + \sum_{\alpha=1}^{\mathsf{d}} R_{i,\alpha} \left[(x - x_i)^\alpha - M_{i,\alpha} \right],$$

  where we have set $M_{i,\alpha} = \frac{1}{h_i} \int_{K_i} (x - x_i)^\alpha \, \mathrm{d}x$ to provide a conservativity property, that is, $\frac{1}{h_i} \int_{K_i} \boldsymbol{\phi}_i(x) \, \mathrm{d}x = \phi_i$, and the vector $R_i = (R_{i,\alpha})_{\alpha=1,\ldots,\mathsf{d}}$ gathers the polynomial coefficients. For a given stencil S_i and vector $\omega_i = (\omega_{i,j})_{j=1,\ldots,\#S_i}$, $\omega_{i,j}$ positive weights of the reconstruction, we consider the quadratic functional

$$\widehat{E}_i(R_i) = \sum_{j \in S_i} \omega_{i,j} \left[\frac{1}{h_j} \int_{K_j} \boldsymbol{\phi}_i(x) \, \mathrm{d}x - \phi_j \right]^2.$$

  We denote by $\widehat{R}_i$ the unique vector which minimizes the quadratic functional and set $\widehat{\boldsymbol{\phi}}_i$ the associated polynomial that corresponds to the best approximation in the least squares sense of the data of the stencil.

- Conservative polynomial reconstruction on boundary interfaces: to take the Dirichlet conditions into account, we introduce two polynomials associated to the boundary interfaces $x = x_{\mathrm{L}}$ and $x = x_{\mathrm{R}}$. The polynomial reconstruction of degree d associated to the boundary interface $x = x_{\mathrm{L}}$ is defined as

$$\boldsymbol{\phi}_{\frac{1}{2}}(x) = \phi_{\mathrm{D}}(x_{\frac{1}{2}}) + \sum_{\alpha=1}^{\mathsf{d}} R_{\frac{1}{2},\alpha} (x - x_{\frac{1}{2}})^\alpha,$$

  where again we have a conservativity property, that is, $\boldsymbol{\phi}_{\frac{1}{2}}(x_{\frac{1}{2}}) = \phi_{\mathrm{D}}(x_{\frac{1}{2}})$, and the vector $R_{\frac{1}{2}} = (R_{\frac{1}{2},\alpha})_{\alpha=1,\ldots,\mathsf{d}}$ gathers the polynomial coefficients. For a given stencil

$S_{\frac{1}{2}}$ and vector $\omega_{\frac{1}{2}} = (\omega_{\frac{1}{2},j})_{j=1,\dots,\#S_{\frac{1}{2}}}$, $\omega_{\frac{1}{2},j}$ positive weights of the reconstruction, we consider the quadratic functional

$$\widehat{E}_{\frac{1}{2}}(R_{\frac{1}{2}}) = \sum_{j \in S_{\frac{1}{2}}} \omega_{\frac{1}{2},j} \left[\frac{1}{h_j} \int_{K_j} \boldsymbol{\phi}_{\frac{1}{2}}(x)\,\mathrm{d}x - \phi_j \right]^2.$$

We denote by $\widehat{R}_{\frac{1}{2}}$ the unique vector which minimizes the quadratic functional and set $\widehat{\boldsymbol{\phi}}_{\frac{1}{2}}$ the associated polynomial that corresponds to the best approximation in the least squares sense of the data of the stencil. We proceed in the same way for polynomial $\widehat{\boldsymbol{\phi}}_{I+\frac{1}{2}}$ at interface $x_{I+\frac{1}{2}} = x_{\mathrm{R}}$.

- Non-conservative polynomial reconstruction at inner interfaces: let $i \in \{1, \dots, I-1\}$. The polynomial reconstruction of degree d associated to interface $x_{i+\frac{1}{2}}$ is defined as

$$\boldsymbol{\phi}_{i+\frac{1}{2}}(x) = \sum_{\alpha=0}^{\mathsf{d}} R_{i+\frac{1}{2},\alpha}(x - x_{i+\frac{1}{2}})^{\alpha},$$

where the vector $R_{i+\frac{1}{2}} = (R_{i+\frac{1}{2},\alpha})_{\alpha=1,\dots,\mathsf{d}}$ gathers the polynomial coefficients. For a given stencil $S_{i+\frac{1}{2}}$ and vector $\omega_{i+\frac{1}{2}} = (\omega_{i+\frac{1}{2},j})_{j=1,\dots,\#S_{i+\frac{1}{2}}}$, $\omega_{i+\frac{1}{2},j}$ positive weights of the reconstruction, we consider the quadratic functional

$$\widetilde{E}_{i+\frac{1}{2}}(R_{i+\frac{1}{2}}) = \sum_{j \in S_{i+\frac{1}{2}}} \omega_{i+\frac{1}{2},j} \left[\frac{1}{h_j} \int_{K_j} \boldsymbol{\phi}_{i+\frac{1}{2}}(x)\,\mathrm{d}x - \phi_j \right]^2.$$

We denote by $\widetilde{R}_{i+\frac{1}{2}}$ the unique vector which minimizes the quadratic functional and set $\widetilde{\boldsymbol{\phi}}_{i+\frac{1}{2}}$ the associated polynomial that corresponds to the best approximation in the least squares sense of the data of the stencil.

2.3. Numerical scheme

To provide an accurate approximation to the convection-diffusion problem, the polynomial reconstructions are employed to design the numerical fluxes. We propose the following expressions based on [4] (we use the notations $[v(x)]^+ = \max(v(x), 0)$ and $[v(x)]^- = \min(v(x), 0)$):

- left boundary interface:

$$\mathcal{F}_{\frac{1}{2}}(\Phi) = \begin{cases} [v(x_{\frac{1}{2}})]^+ \phi_{\mathrm{D}}(x_{\frac{1}{2}}) + [v(x_{\frac{1}{2}})]^- \widehat{\boldsymbol{\phi}}_1(x_{\frac{1}{2}}) - \kappa(x_{\frac{1}{2}}) \widehat{\boldsymbol{\phi}}'_{\frac{1}{2}}(x_{\frac{1}{2}}), & \text{if Dirichlet;} \\ \phi_{\mathrm{T}}(x_{\frac{1}{2}}), & \text{if total Neumann;} \\ v(x_{\frac{1}{2}}) \widehat{\boldsymbol{\phi}}_1(x_{\frac{1}{2}}) + \phi_{\mathrm{P}}(x_{\frac{1}{2}}), & \text{if partial Neumann;} \end{cases}$$

- inner interfaces ($i = 1, \dots, I-1$):

$$\mathcal{F}_{i+\frac{1}{2}}(\Phi) = [v(x_{i+\frac{1}{2}})]^+ \widehat{\boldsymbol{\phi}}_i(x_{i+\frac{1}{2}}) + [v(x_{i+\frac{1}{2}})]^- \widehat{\boldsymbol{\phi}}_{i+1}(x_{i+\frac{1}{2}}) - \kappa(x_{i+\frac{1}{2}}) \widetilde{\boldsymbol{\phi}}'_{i+\frac{1}{2}}(x_{i+\frac{1}{2}});$$

- right boundary interface: similar to the left boundary interface.

Remark 1. *Total Neumann boundary condition at point $\bar{x}$ contains the convective contribution which implicitly provides a Dirichlet-like condition. Indeed, for a zero diffusion process, this type of boundary condition prescribes the convective flux, namely $v(\bar{x})\phi(\bar{x})$. Since $v(\bar{x})$ is given, the $\phi(\bar{x})$ value is implicitely fixed. The problem is that such a situation is not physically relevant for pure convection problem with an outflow condition. When dealing with partial Neumann boundary condition, one has to discretize the convective contribution. Since no value on the boundary $\bar{x}$ is available, the scheme does not involve the upwinding for inflow conditions. It implies that the method is not unconditionally stable with respect to the space parameter. In particular, for an inflow situation, we are dealing with a downwind scheme which requires a stability condition with respect to the velocity, the diffusion, and the characteristic length of the mesh.*

Based on the linearity of the polynomial reconstruction, the definition of the numerical fluxes, and the finite volume formulation (2), we obtain an affine operator such that for any $\Phi \in \mathbb{R}^I$, we associate $\mathcal{G}(\Phi) \in \mathbb{R}^I$ given component-wise by

$$\mathcal{G}_i(\Phi) = \frac{1}{h_i}\left(\mathcal{F}_{i+\frac{1}{2}}(\Phi) - \mathcal{F}_{i-\frac{1}{2}}(\Phi)\right) - f_i,$$

where the affine operator is parameterized by the data of the problem κ, v, f, ϕ_{D}, ϕ_{T}, ϕ_{P}, the mesh, the stencils, and the weights of the quadratic functionals.

The numerical solution is then given by vector $\Phi^\dagger = (\phi_i^\dagger)_{i=1,\dots,I}$ which is the solution of the linear problem $\mathcal{G}(\Phi) = 0_I$.

Remark 2. *Notice that the problem is matrix-free in the sense that we do not explicitly produce the matrix and the right-hand side associated to the affine problem. Iterative methods like Krylov or Jacobi methods take advantage of it.*

3. Time-dependent convection-diffusion problem

We now turn to the time-dependent situation and develop very accurate finite volume schemes both in space and time. We reuse the notation introduced in the previous section, just pointing out the necessary adaptations. For instance, the boundary conditions and the polynomial reconstructions now depend on the time parameter t.

3.1. Finite volume discretization

We seek for function $\phi = \phi(x,t)$ solution of the time-dependent convection-diffusion equation

$$\partial_t\phi + \partial_x(v\phi) - \partial_x(\kappa\partial_x\phi) = f, \quad \text{in } \Omega \times (0, t_{\mathrm{f}}], \tag{3}$$

where $\kappa = \kappa(x,t)$ and $v = v(x,t)$ are regular functions with $\kappa(x,t) \geqslant \kappa_0 > 0$, $f = f(x,t)$ is a regular source term, and $t_{\mathrm{f}} > 0$ is the final time. We also prescribe the initial condition

$$\phi(\cdot, 0) = \phi^0, \quad \text{in } \Omega,$$

and consider three different types of boundary conditions:

- Dirichlet: $\phi(\bar{x},t) = \phi_{\mathrm{D}}(\bar{x},t)$;
- total Neumann: $v(\bar{x},t)\phi(\bar{x},t) - \kappa(\bar{x},t)\partial_x\phi(\bar{x},t) = \phi_{\mathrm{T}}(\bar{x},t)$;
- partial Neumann: $-\kappa(\bar{x},t)\partial_x\phi(\bar{x},t) = \phi_{\mathrm{P}}(\bar{x},t)$,

where again $\bar{x}$ is a boundary point and $\phi_{\mathrm{D}} = \phi_{\mathrm{D}}(x,t)$, $\phi_{\mathrm{T}} = \phi_{\mathrm{T}}(x,t)$, and $\phi_{\mathrm{P}} = \phi_{\mathrm{P}}(x,t)$ are given functions. Of course, compatibility conditions with the initial condition are required to provide the full regularity when dealing with smooth functions.

We apply the method of lines starting by integrating equation (3) over cell K_i to provide the semi-discretization in space

$$\frac{\mathrm{d}\bar{\phi}_i}{\mathrm{d}t}(t) + \frac{1}{h_i}\left(\mathbb{F}_{i+\frac{1}{2}}(t) - \mathbb{F}_{i-\frac{1}{2}}(t)\right) - \bar{f}_i(t) = 0,\ t \in (0, t_{\mathrm{f}}],$$

with

$$\begin{aligned}
\bar{\phi}_i(t) &= \frac{1}{h_i}\int_{K_i} \phi(\xi,t)\,\mathrm{d}\xi,\\
\mathbb{F}_{i\pm\frac{1}{2}}(t) &= v(x_{i\pm\frac{1}{2}},t)\phi(x_{i\pm\frac{1}{2}},t) - \kappa(x_{i\pm\frac{1}{2}},t)\partial_x\phi(x_{i\pm\frac{1}{2}},t),\\
\bar{f}_i(t) &= \frac{1}{h_i}\int_{K_i} f(\xi,t)\,\mathrm{d}\xi.
\end{aligned}$$

3.2. Discretization in space

To derive the space discretization we first introduce vector $\Phi(t) = (\phi_i(t))_{i=1,\ldots,I}$, where $\phi_i(t)$ is an approximation of $\bar{\phi}_i(t)$. We now substitute the exact fluxes $\mathbb{F}_{i\pm\frac{1}{2}}(t)$ on interfaces by the numerical approximations $\mathcal{F}_{i\pm\frac{1}{2}}(t)$. To this end, we consider an extension of the polynomial reconstruction where the time is a parameter. For t fixed, we determine vectors $\widehat{R}_i$, $i = 1,\ldots,I$, $\widehat{R}_{\frac{1}{2}}$, $\widehat{R}_{I+\frac{1}{2}}$, and $\widetilde{R}_{i+\frac{1}{2}}$, $i = 1,\ldots,I-1$, in the same way we have done in the previous section. Thus, we get the time-parameterized polynomial reconstructions $\widehat{\boldsymbol{\phi}}_i(x,t)$, $i = 1,\ldots,I$, $\widehat{\boldsymbol{\phi}}_{\frac{1}{2}}(x,t)$, $\widehat{\boldsymbol{\phi}}_{I+\frac{1}{2}}(x,t)$, and $\widetilde{\boldsymbol{\phi}}_{i+\frac{1}{2}}(x,t)$, $i = 1,\ldots,I-1$. From this polynomial reconstruction extension, we propose the following expressions for the numerical fluxes:

- left boundary interface:

$$\mathcal{F}_{\frac{1}{2}}(t,\Phi(t)) = \begin{cases}
[v(x_{\frac{1}{2}},t)]^+\phi_{\mathrm{D}}(x_{\frac{1}{2}},t) + [v(x_{\frac{1}{2}},t)]^-\widehat{\boldsymbol{\phi}}_1(x_{\frac{1}{2}},t) & \\
\qquad -\kappa(x_{\frac{1}{2}},t)\partial_x\widehat{\boldsymbol{\phi}}_{\frac{1}{2}}(x_{\frac{1}{2}},t), & \text{if Dirichlet;}\\
\phi_{\mathrm{T}}(x_{\frac{1}{2}},t), & \text{if total Neumann;}\\
v(x_{\frac{1}{2}},t)\widehat{\boldsymbol{\phi}}_1(x_{\frac{1}{2}},t) + \phi_{\mathrm{P}}(x_{\frac{1}{2}},t), & \text{if partial Neumann;}
\end{cases}$$

- inner interfaces ($i = 1,\ldots,I-1$):

$$\begin{aligned}
\mathcal{F}_{i+\frac{1}{2}}(t,\Phi(t)) = [v(x_{i+\frac{1}{2}},t)]^+\widehat{\boldsymbol{\phi}}_i(x_{i+\frac{1}{2}},t) + [v(x_{i+\frac{1}{2}},t)]^-\widehat{\boldsymbol{\phi}}_{i+1}(x_{i+\frac{1}{2}},t)&\\
- \kappa(x_{i+\frac{1}{2}},t)\partial_x\widetilde{\boldsymbol{\phi}}_{i+\frac{1}{2}}(x_{i+\frac{1}{2}},t);&
\end{aligned}$$

- right boundary interface: similar to the left boundary interface.

The exact mean source term $\bar{f}_i(t)$ is substituted by the Gaussian quadrature approximation $f_i(t)$ and the residual form of the finite volume scheme on cell K_i writes

$$\frac{\mathrm{d}\phi_i}{\mathrm{d}t}(t) + \mathcal{G}_i(t, \Phi(t)) = 0, \ i = 1, \ldots, I, \tag{4}$$

where

$$\mathcal{G}_i(t, \Phi(t)) = \frac{1}{h_i}\left(\mathcal{F}_{i+\frac{1}{2}}(t, \Phi(t)) - \mathcal{F}_{i-\frac{1}{2}}(t, \Phi(t))\right) - f_i(t).$$

The time parameterized function $\Phi^\dagger(t) = (\phi_i^\dagger(t))_{i=1,\ldots,I}$ is the solution of the differential system (4), which gathers all the components for $t \in (0, t_\mathrm{f}]$ such that

$$\phi_i^\dagger(0) = \frac{1}{h_i}\int_{K_i} \phi^0(\xi)\,\mathrm{d}\xi.$$

The main issue we shall address in the following subsections is that, in the present form, operator $\mathcal{G} = \mathcal{G}(t, \Phi(t))$ does not distinguish the source term and the boundary Dirichlet or Neumann conditions. It results that the time discretization, for instance a Runge-Kutta method, operates in the same way with the three time-dependant contributions which, as we shall highlight, is responsible in some loss of accuracy. Indeed, incompatibilities in the time discretization between the source term and the boundary Dirichlet or Neumann conditions may arise leading to a convergence order degradation of the overall scheme.

3.3. Runge-Kutta methods

To proceed with the time discretization, let N be a positive integer and set the time step $\Delta t = \frac{t_\mathrm{f}}{N}$ and the time subdivision $t^n = n\Delta t$, $n = 0, \ldots, N$. Operator $\mathcal{G} = \mathcal{G}(t, \Phi(t))$ is a time-parameterized function and the generic s-stage Runge-Kutta method to solve the initial value ODE system (4) is given by

$$\Phi^{n+1} = \Phi^n + \Delta t \sum_{k=1}^{s} b_k \mathcal{K}^{n,k}, \tag{5}$$

starting from an initial value vector Φ^0. In the previous equation we have set (notice the minus sign)

$$\mathcal{K}^{n,k} = -\mathcal{G}\left(t^{n,k}, \Phi^n + \Delta t \sum_{\ell=1}^{s} a_{k,\ell}\mathcal{K}^{n,\ell}\right), \ k = 1, \ldots, s, \tag{6}$$

at time $t^{n,k} = t^n + \Delta t c_k$, with $a = (a_{k,\ell})$, an $s \times s$ real matrix, and $b = (b_k)$ and $c = (c_k)$, two vectors of $\mathbb{R}^s$, usually presented in a table called *Butcher Tableau* like Table 1. We recall that we have an explicit Runge-Kutta — ERK if the elements $a_{i,j} = 0$ for $i \leqslant j$; if there exists at least one element $a_{i,j}$ for $i \leqslant j$ that is non-null, then we obtain an implicit Runge-Kutta method — IRK; if $a_{i,j} = 0$ for $i < j$ and $a_{1,1} = 0$ with all diagonal entries equal, we have an explicit singly diagonally implicit Runge-Kutta — ESDIRK). We rewrite the Runge-Kutta formulation (5-6) into an equivalent but more convenient formulation using intermediate stage vectors

$$\Phi^{n,k} = \Phi^n + \Delta t \sum_{\ell=1}^{s} a_{k,\ell}\mathcal{K}^{n,\ell},$$

Table 1: *Butcher Tableau* for the generic s-stage Runge-Kutta method

$$\begin{array}{c|ccc} c_1 & a_{1,1} & \cdots & a_{1,s} \\ \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s,1} & \cdots & a_{s,s} \\ \hline & b_1 & \cdots & b_s \end{array}$$

being the s-stage Runge-Kutta methods given by

$$\Phi^{n,k} - \Phi^n + \Delta t \sum_{\ell=1}^{s} a_{k,\ell} \mathcal{G}(t^{n,\ell}, \Phi^{n,\ell}) = 0_I, \ k = 1, \ldots, s,$$

and

$$\Phi^{n+1} = \Phi^n - \Delta t \sum_{k=1}^{s} b_k \mathcal{G}(t^{n,k}, \Phi^{n,k}).$$

As a consequence, the source term and the boundary conditions are evaluated at the same time $t^{n,k}$ to compute stage $\Phi^{n,k}$ which will bring some numerical errors leading to a loss of accuracy as we show in the two next subsections.

3.4. The Dirichlet case

We first deal with Dirichlet boundary conditions on both boundary points.

3.4.1. Example of accuracy losing for the Dirichlet case

We detail an example to highlight that the interaction between the time discretization and the time-dependent boundary condition may lead to a loss of accuracy. To do so, let consider an example with null velocity since the difficulty is already contained in the diffusive contribution so we assume that the following conditions hold: $\Omega = (0, 1)$, $v(x, t) = 0$, $\kappa(x, t) = 1$, $f(x, t) = 2t$, $\phi^0(x) = 0$, and $\phi_{\mathrm{D}}(\bar{x}, t) = t^2$, where $\bar{x}$ is a boundary point. The solution to this problem is the function $\phi(x, t) = t^2$ and it is well-known that the explicit Runge-Kutta three-stage third-order method (RK3) defined by *Butcher Tableau* presented in Table 2 is exact for polynomial functions up to degree 2. Consequently, the initial value ODE problem $\psi'(t) = 2t$, $\psi(0) = 0$, will be solved exactly. Thus, as ϕ, the solution of the one-dimensional time-dependent diffusion problem we are considering, is constant in space, the diffusion contribution vanishes. Applying the RK3 time discretization with the finite volume scheme presented in section 3.2 even with $\mathsf{d} = 1$ should provide the exact solution.

We now detail each stage of the RK3 method performing one time step from $t^0 = 0$ using a uniform mesh $h = h_i = \frac{1}{I}$, $i = 1, \ldots, I$. Let $\Phi^{n,k} = (\phi_i^{n,k})_{i=1,\ldots,I}$, where $\phi_i^{n,k}$ is an approximation of $\bar{\phi}_i(t^{n,k})$. The numerical fluxes write: $\mathcal{F}_{\frac{1}{2}}(t^{n,k}, \Phi^{n,k}) = -\frac{\phi_1^{n,k} - \phi_{\mathrm{D}}(0, t^{n,k})}{h/2} = -\frac{\phi_1^{n,k} - (t^{n,k})^2}{h/2}$, $\mathcal{F}_{i+\frac{1}{2}}(t^{n,k}, \Phi^{n,k}) = -\frac{\phi_{i+1}^{n,k} - \phi_i^{n,k}}{h}$, $i = 1, \ldots, I-1$, and $\mathcal{F}_{I+\frac{1}{2}}(t^{n,k}, \Phi^{n,k}) = -\frac{\phi_{\mathrm{D}}(1, t^{n,k}) - \phi_I^{n,k}}{h/2} = -\frac{(t^{n,k})^2 - \phi_I^{n,k}}{h/2}$.

Table 2: *Butcher Tableau* for RK3 method

0	0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0
1	-1	2	0
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

- Since $\phi^0(x) = 0$ and $\phi_{\mathrm{D}}(\bar{x}, 0) = 0$, we easily check that at time $t^{0,1} = 0$, we have $\Phi^{0,1} = \Phi^0 = 0_I$ and $\mathcal{G}(t^{0,1}, \Phi^{0,1}) = 0_I$.

- For the second stage, the *Butcher Tableau* yields

$$\Phi^{0,2} - \Phi^0 + \frac{\Delta t}{2}\mathcal{G}(t^{0,1}, \Phi^0) = 0_I,$$

and we deduce that $\Phi^{0,2} = 0_I$. We now compute expression $\mathcal{G}(t^{0,2}, \Phi^{0,2})$ with $t^{0,2} = \frac{\Delta t}{2}$, which implies that $\phi_{\mathrm{D}}(\bar{x}, t^{0,2}) = \frac{\Delta t^2}{4}$ and $f(x, t^{0,2}) = \Delta t$. Since $\phi_{\mathrm{D}}(\bar{x}, t^{0,2}) \neq 0$ whereas $\Phi^{0,2} = 0_I$, **the diffusion contribution is not equal to zero** and we get $\mathcal{F}_{\frac{1}{2}}(t^{0,2}, \Phi^{0,2}) = \frac{\Delta t^2}{2h}$ and $\mathcal{F}_{I+\frac{1}{2}}(t^{0,2}, \Phi^{0,2}) = -\frac{\Delta t^2}{2h}$ on the boundaries. Consequently, on the first cell K_1 and the last cell K_I, the residual values $\mathcal{G}_1(t^{0,2}, \Phi^{0,2})$ and $\mathcal{G}_I(t^{0,2}, \Phi^{0,2})$ correspond to the contribution of the source term and the non-null diffusion fluxes whereas inner cells K_i, $i = 2, \ldots, I-1$, are not affected by the diffusion contribution since $\Phi^{0,2}$ is a constant vector. After calculations, we obtain

$$\mathcal{G}(t^{0,2}, \Phi^{0,2}) = -(\Delta t, \Delta t, \ldots, \Delta t, \Delta t)^T - \left(\frac{\Delta t^2}{2h^2}, 0, \ldots, 0, \frac{\Delta t^2}{2h^2}\right)^T.$$

- We now evaluate the last stage. Since we have

$$\Phi^{0,3} - \Phi^0 - \Delta t\mathcal{G}(t^{0,1}, \Phi^{0,1}) + 2\Delta t\mathcal{G}(t^{0,2}, \Phi^{0,2}) = 0_I,$$

we deduce that

$$\begin{aligned}\Phi^{0,3} &= -2\Delta t\mathcal{G}(t^{0,2}, \Phi^{0,2}) \\ &= \left(2\Delta t^2, 2\Delta t^2, \ldots, 2\Delta t^2, 2\Delta t^2\right)^T + \left(\frac{\Delta t^3}{h^2}, 0, \ldots, 0, \frac{\Delta t^3}{h^2}\right)^T.\end{aligned}$$

To provide $\mathcal{G}(t^{0,3}, \Phi^{0,3})$, we notice that $t^{0,3} = \Delta t$ which yields that $\phi_{\mathrm{D}}(\bar{x}, t^{0,3}) = \Delta t^2$ and $f(x, t^{0,3}) = 2\Delta t$. Both cells K_1 and K_2 are affected by the left boundary condition (and symmetrically cells K_{I-1} and K_I). Fluxes calculation give $\mathcal{F}_{\frac{1}{2}}(t^{0,3}, \Phi^{0,3}) = -\frac{2\Delta t^2}{h} - \frac{2\Delta t^3}{h^3}$, $\mathcal{F}_{\frac{3}{2}}(t^{0,3}, \Phi^{0,3}) = \frac{\Delta t^3}{h^3}$, $\mathcal{F}_{i+\frac{1}{2}}(t^{0,3}, \Phi^{0,3}) = 0$, $i = 2, \ldots, I-2$, $\mathcal{F}_{I-\frac{1}{2}}(t^{0,3}, \Phi^{0,3}) = -\frac{\Delta t^3}{h^3}$, and $\mathcal{F}_{I+\frac{1}{2}}(t^{0,3}, \Phi^{0,3}) = \frac{2\Delta t^2}{h} + \frac{2\Delta t^3}{h^3}$. After some algebraic manipulations we get:

$$\begin{aligned}\mathcal{G}(t^{0,3}, \Phi^{0,3}) = &(-2\Delta t, -2\Delta t, -2\Delta t, \ldots, -2\Delta t, -2\Delta t, -2\Delta t)^T \\ &+ \left(2\frac{\Delta t^2}{h^2} + 3\frac{\Delta t^3}{h^4}, -\frac{\Delta t^3}{h^4}, 0, \ldots, 0, -\frac{\Delta t^3}{h^4}, 2\frac{\Delta t^2}{h^2} + 3\frac{\Delta t^3}{h^4}\right)^T.\end{aligned}$$

- At last, we compute the approximations at time $t^1 = \Delta t$ with

$$\Phi^1 = \Phi^0 - \frac{\Delta t}{6}\mathcal{G}(t^{0,1}, \Phi^{0,1}) - \frac{2\Delta t}{3}\mathcal{G}(t^{0,2}, \Phi^{0,2}) - \frac{\Delta t}{6}\mathcal{G}(t^{0,3}, \Phi^{0,3}),$$

obtaining

$$\Phi^1 = \left(\Delta t^2, \Delta t^2, \Delta t^2, \ldots, \Delta t^2, \Delta t^2, \Delta t^2\right)^T + \left(-\frac{\Delta t^4}{2h^4}, \frac{\Delta t^4}{6h^4}, 0, \ldots, 0, \frac{\Delta t^4}{6h^4}, -\frac{\Delta t^4}{2h^4}\right)^T.$$

The solution at the first time step does not present the correct values on the cells close to the boundaries and the solution is no longer constant in space. Numerical perturbation appears due to the fact that the values deriving from the boundary conditions discretization do not match with the ones deriving from the equation discretization. Error $\frac{\Delta t^4}{h^4}$ is generated at the very beginning of the computation. High-order reconstructions may reduce such phenomenon but will not cure it since the fundamental problem remains: the source term discretization does not match the Dirichlet discretization in time.

3.4.2. Extended Butcher Tableau for the Dirichlet conditions

To overcome the incompatibility problem, we shall consider a relaxation of the discretization considering now that the time discretizations for the source term and for the Dirichlet condition can be different. Let us define

$$\Phi_{\mathrm{D}}^{n,k} = \begin{pmatrix} \phi_{\mathrm{D}}(x_{\mathrm{L}}, t^{n,k}) \\ \phi_{\mathrm{D}}(x_{\mathrm{R}}, t^{n,k}) \end{pmatrix}.$$

We re-qualify the residual operator setting

$$\mathcal{G}^{n,k} = \mathcal{G}(t^{n,k}, \Phi^{n,k}, F^{n,k}, \Phi_{\mathrm{D}\star}^{n,k}),$$

where $F^{n,k} = (f_i(t^{n,k}))_{i=1,\ldots,I}$ and vector $\Phi_{\mathrm{D}\star}^{n,k}$, the substitute of $\Phi_{\mathrm{D}}^{n,k}$, should be evaluated/discretized in a different way. We here introduce a generic principle to derive the two discretizations:

> **Principle:** The time discretizations of the source term and the Dirichlet condition are compatible up to degree d if the scheme exactly solves the solutions t^m, $m = 0, \ldots, d$, constant in space.

Such a principle may not be fulfilled if one considers whatever time and space dependent convection or diffusion coefficients. Therefore, we restrict the principle to a very simple situation taking $\kappa = 1$, $v = 0$ and build the extension for this particular case. Hence we shall check in the numerical section that the method based on the extended tableau significantly increases the accuracy even for complex situations with non-constant coefficients.

Since we are now dealing with constant in space functions, the diffusive contribution vanishes. Therefore, we naturally introduce the initial value Ordinary Differential Equation $\psi'(t) = g(t)$, $\psi(0) = \psi^0 \in \mathbb{R}$ where we shall take the functions t^m, $m = 0, \ldots, d$, for

ψ. Based on the *Butcher Tableau*, the Runge-Kutta method for the initial value ODE problem is governed by the linear relations

$$\psi^{n,k} = \psi^n + \Delta t \sum_{\ell=1}^{s} a_{k,\ell} g(t^{n,\ell}), k = 1, \ldots, s,$$

and

$$\psi^{n+1} = \psi^n + \Delta t \sum_{k=1}^{s} b_k g(t^{n,k}),$$

since g does not depend on ψ.

The choice of degree m is directly linked to the Runge-Kutta method order to provide the optimal convergence. For the sake of simplicity we assume that we use an s-stage RK method for an ODE which exactly solve all the polynomial solutions up to degree $s-1$, *i.e.* $m = s - 1$ (the more general case will be tackled in the examples section). The principle we introduce states that we shall recover the same convergence order when dealing with a constant in space solution for the convection-diffusion problem. We highlight that an s-order in time for constant in space solution and a p-order in space for constant in time solution method does not necessarily solve in an exact way monomial functions of the form $x^\sigma t^\tau$, $\tau < s$, $\sigma < p$. Hence the global error may be lower than $\min(s, p)$. Nevertheless, the principle reveals the origin of the error: in the RK method, the solution at the different substeps does not correspond to the intermediate approximations, only the final combination provides the good approximation.

To reproduce constant is space solutions for the one-dimensional convection-diffusion problem, the key idea is to define a boundary condition discretization $\Phi_{\mathrm{D}\star}^{n,k}$ which is not necessarily $\Phi_{\mathrm{D}}^{n,k}$ as we have done in the previous paragraph but to consider a linear combination of the different time stages, namely

$$\Phi_{\mathrm{D}\star}^{n,k} = \sum_{\ell=1}^{s} p_{k,\ell} \Phi_{\mathrm{D}}^{n,\ell}, \ k = 1, \ldots, s.$$

To determine the $s \times s$ matrix $p = (p_{k,\ell})$, we apply the principle setting $\psi(t) = \phi_{\mathrm{D}}(x, t) = t^m$, $m = 0, \ldots, d$, such that the vectorial equality

$$\Phi_{\mathrm{D}\star}^{n,k} = \psi^{n,k} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \ k = 1, \ldots, s \tag{7}$$

holds. Since the solution is constant in space, the vectorial equality (7) turns out to be a scalar one. Therefore, only the first component is relevant. In other words, $\Phi_{\mathrm{D}\star}^{n,k}$ is evaluated such that the values in the domain (governed by Runge-Kutta method) exactly match the values at the boundary for a class of polynomial functions.

To effectively determine matrix p entries, let $\bar{t}$ be a reference time. As we can always shift the time referential, one can choose $\bar{t} = 0$ and the time discretization writes

$$\sum_{\ell=1}^{s} p_{k,\ell} = 1, \ k = 1, \ldots, s,$$

for $m = 0$ and

$$\sum_{\ell=1}^{s} p_{k,\ell}(c_\ell \Delta t)^m = \Delta t \sum_{\ell=1}^{s} a_{k,\ell} m (c_\ell \Delta t)^{m-1},\ k = 1, \ldots, s,$$

for $m = 1, \ldots, d$. Let us consider the Vandermonde matrices

$$Q^d = \begin{pmatrix} (c_1)^0 & (c_2)^0 & \cdots & (c_s)^0 \\ (c_1)^1 & (c_2)^1 & \cdots & (c_s)^1 \\ (c_1)^2 & (c_2)^2 & \cdots & (c_s)^2 \\ \vdots & \vdots & \ddots & \vdots \\ (c_1)^{d-1} & (c_2)^{d-1} & \cdots & (c_s)^{d-1} \\ (c_1)^d & (c_2)^d & \cdots & (c_s)^d \end{pmatrix}$$

and

$$\widehat{Q}^d = \begin{pmatrix} 0 & 0 & \ldots & 0 \\ 1(c_1)^0 & 1(c_2)^0 & \cdots & 1(c_s)^0 \\ 2(c_1)^1 & 2(c_2)^1 & \cdots & 2(c_s)^1 \\ \vdots & \vdots & \ddots & \vdots \\ (d-1)(c_1)^{d-2} & (d-1)(c_2)^{d-2} & \cdots & (d-1)(c_s)^{d-2} \\ d(c_1)^{d-1} & d(c_2)^{d-1} & \cdots & d(c_s)^{d-1} \end{pmatrix}.$$

Then the condition writes for $k = 1, \ldots, s$,

$$Q^d \begin{pmatrix} p_{k,1} \\ p_{k,2} \\ p_{k,3} \\ \vdots \\ p_{k,s-1} \\ p_{k,s} \end{pmatrix} = \widehat{Q}^d \begin{pmatrix} a_{k,1} \\ a_{k,2} \\ a_{k,3} \\ \vdots \\ a_{k,s-1} \\ a_{k,s} \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix},$$

which provides the row k of matrix p. In compact form, one has $Q^d p^T = \widehat{Q}^d a^T + E_1^{d+1,s}$, where $E_1^{d+1,s}$ is a $(d+1) \times s$ matrix, such that the elements of the first row are ones and all the other elements are zeros. We assume that coefficients c_ℓ, $\ell = 1, \ldots, s$, are different from each others for the sake of simplicity but the case when two coefficients are equal can be also treated as proposed in section 3.7. If $s = d + 1$, the Vandermonde matrix Q^d is square and invertible while $s > d + 1$ corresponds to an under-determined linear system associated to a maximal rank matrix Q^d. In the first case, existence and uniqueness of matrix p is straightforward but for the second case one has to develop a strategy to determine a unique matrix p. We here propose two ways to determine matrix p namely:

- LS-way: find p in the Least Square sense;
- AC-way: Augment the number of Constraints adding polynomial functions t^{d+1}, ..., t^{s-1} such that we get a invertible Vandermonde square matrix Q^{s-1}.

From the *Extended Butcher Tableau*, we derived the new Runge-Kutta formulation for the Dirichlet-Dirichlet problem: find the $\Phi^{n,k}$, $k = 1, \ldots, s$, such that

$$\Phi^{n,k} - \Phi^n + \Delta t \sum_{\ell=1}^{s} a_{k,\ell} \mathcal{G}(t^{n,\ell}, \Phi^{n,\ell}, F^{n,\ell}, \Phi_{\mathrm{D}\star}^{n,\ell}) = 0_I$$

and compute

$$\Phi^{n+1} = \Phi^n - \Delta t \sum_{k=1}^{s} b_k \mathcal{G}(t^{n,k}, \Phi^{n,k}, F^{n,k}, \Phi_{\mathrm{D}\star}^{n,k})$$

where we have set

$$\Phi_{\mathrm{D}\star}^{n,\ell} = \sum_{\ell=1}^{s} p_{k,\ell} \Phi_{\mathrm{D}}^{n,\ell}.$$

In Table 3 we represent in compact form the data needed for the presented extension.

Table 3: *Extended Butcher Tableau* for the generic s-stage Runge-Kutta method

c_1	$a_{1,1}$	$\cdots$	$a_{1,s}$	$p_{1,1}$	$\cdots$	$p_{1,s}$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
c_s	$a_{s,1}$	$\cdots$	$a_{s,s}$	$p_{s,1}$	$\cdots$	$p_{s,s}$
	b_1	$\cdots$	b_s			

3.5. The Neumann case

We now turn to the situation where we have a total Neumann boundary condition on both boundary points.

3.5.1. Example of accuracy loss for the Neumann case

We here give an example to highlight the incompatibility between the the source term discretization and the Neumann condition. We again consider a pure diffusive and data of the test case are the followings: $\Omega = (0,1)$, $v(x,t) = 0$, $\kappa(x,t) = 1$, $f(x,t) = 2xt$, $\phi^0(x) = 0$, and $\phi_{\mathrm{T}}(\bar{x}, t) = -t^2$, where $\bar{x}$ is a boundary point. One easily checks that the solution is function $\phi(x,t) = xt^2$.

Let $\psi = \psi(t)$ be the solution of the initial value ODE problem $\psi'(t) = 2t$, $\psi(0) = 0$. Since the source term is a linear function, again the RK3 scheme exactly solves this problem. Therefore, noticing that ϕ, the solution of the one-dimensional time-dependent diffusion problem we are considering, satisfies the relation $\phi(x,t) = x\psi(t)$, we expect that the RK3 time scheme applied to the ϕ problem would also produce the exact solution.

As for the Dirichlet case, we detail each stage of the RK3 method performing one time step from $t^0 = 0$ using a uniform mesh $h = h_i = \frac{1}{I}$, $i = 1, \ldots, I$. Let $\Phi^{n,k} = (\phi_i^{n,k})_{i=1,\ldots,I}$, where $\phi_i^{n,k}$ is an approximation of $\bar{\phi}_i(t^{n,k})$. The numerical fluxes write $\mathcal{F}_{\frac{1}{2}}(t^{n,k}, \Phi^{n,k}) = -\phi_{\mathrm{T}}(0, t^{n,k}) = -(t^{n,k})^2$, $\mathcal{F}_{i+\frac{1}{2}}(t^{n,k}, \Phi^{n,k}) = -\frac{\phi_{i+1}^{n,k} - \phi_i^{n,k}}{h}$, $i = 1, \ldots, I-1$, $\mathcal{F}_{I+\frac{1}{2}}(t^{n,k}, \Phi^{n,k}) = -\phi_{\mathrm{T}}(1, t^{n,k}) = -(t^{n,k})^2$.

- Since $\phi^0(x) = 0$ and $\phi_{\mathrm{T}}(\bar{x}, 0) = 0$, we easily check that at time $t^{0,1} = 0$ we have $\Phi^0 = \Phi^{0,1} = 0_I$ and $\mathcal{G}(t^{0,1}, \Phi^{0,1}) = 0_I$.

- For the second stage, the *Butcher Tableau* yields

$$\Phi^{0,2} - \Phi^0 + \frac{\Delta t}{2}\mathcal{G}(t^{0,1}, \Phi^0) = 0_I$$

and we deduce that $\Phi^{0,2} = 0_I$. We now compute expression $\mathcal{G}(t^{0,2}, \Phi^{0,2})$ with $t^{0,2} = \frac{\Delta t}{2}$, which implies that $\phi_{\mathrm{T}}(\bar{x}, t^{0,2}) = \frac{\Delta t^2}{4}$, and $f(x, t^{0,2}) = x\Delta t$. The fluxes at the interfaces are $\mathcal{F}_{\frac{1}{2}}(t^{0,2}, \Phi^{0,2}) = -\frac{\Delta t^2}{4}$, $\mathcal{F}_{i+\frac{1}{2}}(t^{0,2}, \Phi^{0,2}) = 0$, $i = 1, \ldots, I-1$, and $\mathcal{F}_{I+\frac{1}{2}}(t^{0,2}, \Phi^{0,2}) = -\frac{\Delta t^2}{4}$. After calculations, we obtain

$$\mathcal{G}(t^{0,2}, \Phi^{0,2}) = -\left(x_1\Delta t, x_2\Delta t, \ldots, x_{I-1}\Delta t, x_I\Delta t\right)^T + \left(\frac{\Delta t^2}{4h}, 0, \ldots, 0, -\frac{\Delta t^2}{4h}\right)^T.$$

- At last, we compute the third stage. Using

$$\Phi^{0,3} - \Phi^0 - \Delta t\mathcal{G}(t^{0,1}, \Phi^{0,1}) + 2\Delta t\mathcal{G}(t^{0,2}, \Phi^{0,2}) = 0_I,$$

we deduce that

$$\begin{aligned}\Phi^{0,3} &= -2\Delta t\mathcal{G}(t^{0,2}, \Phi^{0,2}) \\ &= \left(2x_1\Delta t^2, 2x_2\Delta t^2, \ldots, 2x_{I-1}\Delta t^2, 2x_I\Delta t^2\right)^T + \left(-\frac{\Delta t^3}{2h}, 0, \ldots, 0, \frac{\Delta t^3}{2h}\right)^T.\end{aligned}$$

With $t^{0,3} = \Delta t$, we have $f(x, t^{0,3}) = 2x\Delta t$, $\phi_{\mathrm{T}}(\bar{x}, t^{0,3}) = \Delta t^2$, and obtain $\mathcal{F}_{\frac{1}{2}}(t^{0,3}, \Phi^{0,3}) = -\Delta t^2$, $\mathcal{F}_{\frac{3}{2}}(t^{0,3}, \Phi^{0,3}) = -2\Delta t^2 - \frac{\Delta t^3}{2h^2}$, $\mathcal{F}_{i+\frac{1}{2}}(t^{0,3}, \Phi^{0,3}) = -2\Delta t^2$, $i = 2, \ldots, I-2$, $\mathcal{F}_{I-\frac{1}{2}}(t^{0,3}, \Phi^{0,3}) = -2\Delta t^2 - \frac{\Delta t^3}{2h^2}$, $\mathcal{F}_{I+\frac{1}{2}}(t^{0,3}, \Phi^{0,3}) = -\Delta t^2$. We then deduce

$$\begin{aligned}\mathcal{G}(t^{0,3}, \Phi^{0,3}) = &-(2x_1\Delta t, 2x_2\Delta t, 2x_3\Delta t, \ldots, 2x_{I-2}\Delta t, 2x_{I-1}\Delta t, 2x_I\Delta t) \\ &+ \left(-\frac{\Delta t^2}{h} - \frac{\Delta t^3}{2h^3}, \frac{\Delta t^3}{2h^3}, \ldots, -\frac{\Delta t^3}{2h^3}, \frac{\Delta t^2}{h} + \frac{\Delta t^3}{2h^3}\right)^T.\end{aligned}$$

- We compute the approximations at time $t^1 = \Delta t$ with

$$\Phi^1 = \Phi^0 - \frac{\Delta t}{6}\mathcal{G}(t^{0,1}, \Phi^{0,1}) - \frac{2\Delta t}{3}\mathcal{G}(t^{0,2}, \Phi^{0,2}) - \frac{\Delta t}{6}\mathcal{G}(t^{0,3}, \Phi^{0,3})$$

and we obtain

$$\begin{aligned}\Phi^1 = &\left(x_1\Delta t^2, x_2\Delta t^2, x_3\Delta t^2, \ldots, x_{I-2}\Delta t^2, x_{I-1}\Delta t^2, x_I\Delta t^2\right)^T \\ &+ \left(\frac{\Delta t^4}{12h^3}, -\frac{\Delta t^4}{12h^3}, 0, \ldots, 0, \frac{\Delta t^4}{12h^3}, -\frac{\Delta t^4}{12h^3}\right)^T.\end{aligned}$$

We get this time an error controlled by $\frac{\Delta t^4}{h^3}$.

3.5.2. Extended Butcher Tableau for the Neumann conditions

The principle proposed in the Dirichlet case can not be here applied since a constant in space function gives rise to homogeneous Neumann conditions. Therefore, we introduce a slightly different generic principle to derive the two discretizations:

> **Principle:** The time discretizations of the source term and the Neumann conditions are compatible up to degree d if the scheme exactly solves the solutions xt^m, $m = 0, \ldots, d$, linear in space.

As in the Dirichlet case, the criterion may not be fulfilled with non-constant coefficients hence we restrict the criterion to the simple situation $\kappa = 1$, $v = 0$ to build the extended Butcher tableau. We numerically check that the method deriving from the extension effectively improves the accuracy. To this end, let us define

$$\Phi_{\mathrm{T}}^{n,k} = \begin{pmatrix} \phi_{\mathrm{T}}(x_{\mathrm{L}}, t^{n,k}) \\ \phi_{\mathrm{T}}(x_{\mathrm{R}}, t^{n,k}) \end{pmatrix}.$$

We consider the residual operator for Neumann conditions setting

$$\mathcal{G}^{n,k} = \mathcal{G}(t^{n,k}, \Phi^{n,k}, F^{n,k}, \Phi_{\mathrm{T}\star}^{n,k}),$$

where again $F^{n,k} = (f_i(t^{n,k}))_{i=1,\ldots,I}$ and vector $\Phi_{\mathrm{T}\star}^{n,k}$, the substitute of $\Phi_{\mathrm{T}}^{n,k}$, will be defined in the sequel. For null velocity, constant diffusion κ, and functions $\phi(x,t) = xt^m$, $m = 0, \ldots, d$, we have

$$\partial_t(xt^m) - \kappa\partial_{xx}(xt^m) = mxt^{m-1} =: f(x,t).$$

Consequently, if $\psi = \psi(t)$ is the solution of $\psi'(t) = mt^{m-1} =: g(t)$, with $\psi(0) = 1$ if $m = 0$ and $\psi(0) = 0$ if $m > 0$, then $\phi = x\psi$ is the solution of the heat equation

$$\partial_t\phi - \kappa\partial_{xx}\phi = f(x,t)(= xg(t)),$$

with the initial condition $\phi^0(x) = x$ if $m = 0$ and $\phi^0(x) = 0$ if $m > 0$ and the Neumann condition $\phi_{\mathrm{T}}(x,t) = -\kappa t^m$.

As in the Dirichlet case, we consider a linear combination of the different time stages, namely

$$\Phi_{\mathrm{T}\star}^{n,k} = \sum_{\ell=1}^{s} q_{k,\ell}\Phi_{\mathrm{T}}^{n,\ell},\ k = 1, \ldots, s,$$

where one has to determine an $s \times s$ matrix $q = (q_{k,\ell})$. Noticing that $\psi = \partial_x(x\psi)$ and $\phi_{\mathrm{T}}(x,t) = -\kappa t^m = -\kappa\phi_{\mathrm{D}}(x,t)$ where ϕ_{D} is the Dirichlet function considered in section 3.4.2, the principle for Neumann condition yields

$$\Phi_{\mathrm{T}\star}^{n,k} = -\kappa\Phi_{\mathrm{D}\star}^{n,k} = -\kappa\psi^{n,k}\begin{pmatrix} 1 \\ 1 \end{pmatrix}, k = 1, \ldots, s. \tag{8}$$

We deduce that relation (7) is identical to relation (8), hence $q = p$. Consequently, we can use the same time discretization both for the Dirichlet and the Neumann conditions characterized by the *Extended Butcher Tableau*.

3.6. Time discretization with Dirichlet and Neumann boundary conditions

We now generalize the extended Runge-Kutta formulation for the Dirichlet-Neuman problem. Let us consider the *Extended Butcher Tableau* given in Table 3 and set $\Phi_{\rm D}^{n,k} = \phi_{\rm D}(x_{\rm L}, t^{n,k})$, $\Phi_{\rm T}^{n,k} = \phi_{\rm T}(x_{\rm R}, t^{n,k})$. Assume that Φ^n is a given approximation of the solution at time t^n. We seek vectors $\Phi^{n,k}$, $k = 1, \ldots, s$, solution of

$$\Phi^{n,k} - \Phi^n + \Delta t \sum_{\ell=1}^{s} a_{k,\ell} \mathcal{G}(t^{n,\ell}, \Phi^{n,\ell}, F^{n,\ell}, \Phi_{\rm D\star}^{n,\ell}, \Phi_{\rm T\star}^{n,\ell}) = 0_I.$$

and we determine

$$\Phi^{n+1} = \Phi^n - \Delta t \sum_{k=1}^{s} b_k \mathcal{G}(t^{n,k}, \Phi^{n,k}, F^{n,k}, \Phi_{\rm D\star}^{n,k}, \Phi_{\rm T\star}^{n,k}),$$

where we have set

$$\Phi_{\rm D\star}^{n,\ell} = \sum_{\ell=1}^{s} p_{k,\ell} \Phi_{\rm D}^{n,\ell}, \ \Phi_{\rm T\star}^{n,\ell} = \sum_{\ell=1}^{s} p_{k,\ell} \Phi_{\rm T}^{n,\ell}.$$

Remark 3. *Extension for higher dimension problems can be considered where vectors $\Phi_D^{n,\ell}$ and $\Phi_T^{n,\ell}$ correspond to the evaluation of the Dirichlet and Neumann conditions at the Gauss points on edges or faces.*

3.7. Extended Butcher Tableaux examples

We present the construction of *Extended Butcher Tableaux* for different orders. We also propose a technique to evaluate matrix p even if two time stages (or more) are equal.

- **Examples of second-order time discretization.**

  For two-stage methods we have the matrices

$$Q^2 = \begin{pmatrix} 1 & 1 \\ c_1 & c_2 \end{pmatrix}, \qquad \widehat{Q}^2 = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}, \qquad E_1^{2,2} = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

  In Table 4 we present three second-order two-stage methods: the Midpoint method (explicit), the Heun method (explicit), and the Crank-Nicholson method (an implicit unconditionally stable discretization).

Table 4: *Extended Butcher Tableaux* for second-order Runge-Kutta methods: Midpoint method (left), Heun method (center), and Crank-Nicholson method (right).

0	0	0	1	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0	1
	0	1		

0	0	0	1	0
1	1	0	0	1
	$\frac{1}{2}$	$\frac{1}{2}$		

0	0	0	1	0
1	$\frac{1}{2}$	$\frac{1}{2}$	0	1
	$\frac{1}{2}$	$\frac{1}{2}$		

  For all these examples we notice that the source term time discretization and the boundary conditions time discretization are equal since p is the identity matrix. Therefore no specific treatment is required for the Boundary condition.

- **Example of third-order time discretization.**

  Let us consider the RK3 given by Table 2 with $s = 3$ and $d = 2$. For three-stage methods we have the matrices

$$Q^3 = \begin{pmatrix} 1 & 1 & 1 \\ c_1 & c_2 & c_3 \\ (c_1)^2 & (c_2)^2 & (c_3)^2 \end{pmatrix}, \quad \widehat{Q}^3 = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 2c_1 & 2c_2 & 2c_3 \end{pmatrix}, \quad E_1^{3,3} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

  The linear system to compute matrix p writes as

$$\begin{pmatrix} 1 & 1 & 1 \\ 0 & \frac{1}{2} & 1 \\ 0 & \frac{1}{4} & 1 \end{pmatrix} \begin{pmatrix} p_{1,1} & p_{2,1} & p_{3,1} \\ p_{1,2} & p_{2,2} & p_{3,2} \\ p_{1,3} & p_{2,3} & p_{3,3} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} 0 & \frac{1}{2} & -1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

  where we use the transpose matrices of p and a. Solving the linear system, we then deduce Table 5. Since matrix p is different to the identity one, specific boundary

Table 5: *Extended Butcher Tableau* for RK3

0	0	0	0	1	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0	$-\frac{1}{2}$	2	$-\frac{1}{2}$
1	-1	2	0	2	-4	3
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$			

  treatment is required to achieve an optimal convergence order.

- **Example of fourth-order time discretization with two equal time stage.**

  Let us consider the explicit 4-stage RK4. Since $c_2 = c_3$, we can not directly apply the methodology developed in the previous section because matrix Q^4 is singular. To overcome the problem, we introduce a new vector $z = (z_1, z_2, z_3, z_4)^T$ and new time stages $\tau^{n,k} = t^n + z_k \Delta t$, $k = 1, \ldots, s$. We present the proof for the Dirichlet situation since the Neumann case will provide the same matrix p. We then define

$$\Phi_{\mathrm{D}\star}^{n,k} = \sum_{\ell=1}^{s} p_{k,\ell} \Phi_{\mathrm{D}}^{n,\ell}, \ k = 1, \ldots, s,$$

  where we have to take, for $m = 0, \ldots, d$,

$$\Phi_{\mathrm{D}}^{n,\ell} = (\tau^{n,\ell})^m \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

  Applying the principle that the source term discretization matches with the Dirichlet condition and taking $t^n = 0$ as the reference time, we get the matrices

$$Q^4 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ z_1 & z_2 & z_3 & z_4 \\ (z_1)^2 & (z_2)^2 & (z_3)^2 & (z_4)^2 \\ (z_1)^3 & (z_2)^3 & (z_3)^3 & (z_4)^3 \end{pmatrix}, \quad \widehat{Q}^4 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 2c_1 & 2c_2 & 2c_3 & 2c_4 \\ 3(c_1)^2 & 3(c_2)^2 & 3(c_3)^2 & 3(c_4)^2 \end{pmatrix}.$$

It is important to notice that matrix Q^4 involves the new coefficients z while matrix $\widehat{Q}^4$ still uses coefficients c. We choose arbitrary $z = (0, \frac{1}{2}, \frac{3}{4}, 1)$ and we get the *Extended Butcher Tableau* in Table 6 where vector z has been stored on the right-hand side of the Tableau.

Table 6: *Extended Butcher Tableau* for RK4 method

0	0	0	0	0	1	0	0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0	0	$-\frac{7}{6}$	6	$-\frac{16}{3}$	$\frac{3}{2}$	$\frac{1}{2}$
$\frac{1}{2}$	0	$\frac{1}{2}$	0	0	$\frac{5}{6}$	-2	$\frac{8}{3}$	$-\frac{1}{2}$	$\frac{3}{4}$
1	0	0	1	0	$\frac{2}{3}$	-4	$\frac{16}{3}$	-1	1
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$					

- **Example of fourth-order time discretization with $s > d + 1$.**

  All the previous cases correspond to the situation where $s = d + 1$. We now consider an implicit version (DIRK) which requires 6 stages, hence $s > d + 1$. As we mentioned in the previous section, the linear system is under-determined and we here propose two strategies: the Least Square method and the Augmented Constraint method.

  For the Least square method, the matrices write

$$Q^4 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ c_1 & c_2 & c_3 & c_4 & c_5 & c_6 \\ (c_1)^2 & (c_2)^2 & (c_3)^2 & (c_4)^2 & (c_5)^2 & (c_6)^2 \\ (c_1)^3 & (c_2)^3 & (c_3)^3 & (c_4)^3 & (c_5)^3 & (c_6)^3 \end{pmatrix}$$

  and

$$\widehat{Q}^4 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 2c_1 & 2c_2 & 2c_3 & 2c_4 & 2c_5 & 2c_6 \\ 3(c_1)^2 & 3(c_2)^2 & 3(c_3)^2 & 3(c_4)^2 & 3(c_5)^2 & 3(c_6)^2 \end{pmatrix}$$

  and we solve the linear system $Q^4 p^T = \widehat{Q}^4 a^T + E_1^{4,6}$ in the least square sense. We present in Table 7 the resulting *Extended Butcher Tableau*.

  A second way consists in augmenting the numbers of constraints assuming that the

Table 7: *Extended Butcher Tableau* for ESDIRK4 method — Least Squares approach

0	0	0	0	0	0	0	1	$-\frac{738}{2093}$	$-\frac{193}{9366}$	$-\frac{151}{1016}$	$-\frac{490}{1663}$	0
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	0	0	0	0	0	0	0	0	0	0
$\frac{83}{250}$	$\frac{8611}{62500}$	$-\frac{1743}{31250}$	$\frac{1}{4}$	0	0	0	0	$\frac{895}{618}$	$\frac{1654}{1551}$	$\frac{501}{1046}$	$\frac{1703}{2211}$	0
$\frac{31}{50}$	$\frac{5012029}{34652500}$	$-\frac{654441}{2922500}$	$\frac{174375}{388108}$	$\frac{1}{4}$	0	0	0	$-\frac{455}{1506}$	$-\frac{319}{5103}$	$\frac{922}{1679}$	$-\frac{805}{4378}$	0
$\frac{17}{20}$	$\frac{15267082809}{155376265600}$	$-\frac{71443401}{120774400}$	$\frac{730878875}{902184768}$	$\frac{2285395}{8070912}$	$\frac{1}{4}$	0	0	0	0	0	0	0
1	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$-\frac{2260}{8211}$	$\frac{1}{4}$	0	$\frac{165}{799}$	$\frac{303}{18133}$	$\frac{400}{3319}$	$\frac{573}{809}$	1
	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$-\frac{2260}{8211}$	$\frac{1}{4}$						

two discretizations also match for t^4 and t^5. In that case the matrices write

$$Q^6 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ c_1 & c_2 & c_3 & c_4 & c_5 & c_6 \\ (c_1)^2 & (c_2)^2 & (c_3)^2 & (c_4)^2 & (c_5)^2 & (c_6)^2 \\ (c_1)^3 & (c_2)^3 & (c_3)^3 & (c_4)^3 & (c_5)^3 & (c_6)^3 \\ (c_1)^4 & (c_2)^4 & (c_3)^4 & (c_4)^4 & (c_5)^4 & (c_6)^4 \\ (c_1)^5 & (c_2)^5 & (c_3)^5 & (c_4)^5 & (c_5)^5 & (c_6)^5 \end{pmatrix}$$

and

$$\widehat{Q}^6 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 2c_1 & 2c_2 & 2c_3 & 2c_4 & 2c_5 & 2c_6 \\ 3(c_1)^2 & 3(c_2)^2 & 3(c_3)^2 & 3(c_4)^2 & 3(c_5)^2 & 3(c_6)^2 \\ 4(c_1)^3 & 4(c_2)^3 & 4(c_3)^3 & 4(c_4)^3 & 4(c_5)^3 & 4(c_6)^3 \\ 5(c_1)^4 & 5(c_2)^4 & 5(c_3)^4 & 5(c_4)^4 & 5(c_5)^4 & 5(c_6)^4 \end{pmatrix},$$

and we solve the linear system $Q^6 p^T = \widehat{Q}^6 a^T + E_1^{6,6}$. We present in Table 8 the resulting *Extended Butcher Tableau*.

4. Numerical tests

At a first stage, we consider the steady-state situation and show that the discretization in space provides an effective sixth-order scheme when dealing with low and large Péclet numbers. We also address the stability question of the total or partial Neumann conditions for the convection-diffusion with large Péclet number. The second part is dedica-ted to the time-dependent equation where we numerically confirm that the *Extended Butcher Tableau* is a necessary ingredient to significantly improve the effective order both in space and time.

The notation $\mathbb{P}_{\mathsf{d}}(n)$ means that the polynomial reconstructions have degree d using stencils of n elements. The weights we will consider are summarized with the notation $\omega_{i,j} = q|r$, $q, r \in \mathbb{R}^+$, with the following meaning: if i and j are contiguous cells, then $\omega_{i,j} = q$; otherwise, $\omega_{i,j} = r$ (this notation extends in the natural way for the cases $\omega_{\frac{1}{2},j}$, $\omega_{i+\frac{1}{2},j}$, and $\omega_{I+\frac{1}{2},j}$). In the present study, all the computations have been carried out with weights $2|1$ and a study on the weights choice is given in [4]. Moreover, to reduce the computational effort, a preconditioning matrix is used as proposed in [4] for the steady-state problems as well as when dealing with time-dependent implicit schemes.

Table 8: *Extended Butcher Tableau* for ESDIRK4 method — Augmented Constraints approach

0	0	0	0	0	0	0	1	$-\frac{1958}{1645}$	$-\frac{346}{1317}$	$-\frac{2585}{6863}$	$-\frac{177}{4970}$	$-\frac{417}{797}$
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	0	0	0	0	0	$-\frac{7388}{283}$	$-\frac{4924}{567}$	$-\frac{2637}{379}$	$\frac{5466}{667}$	$-\frac{4956}{191}$
$\frac{83}{250}$	$\frac{8611}{62500}$	$-\frac{1743}{31250}$	$\frac{1}{4}$	0	0	0	0	$\frac{9991}{878}$	$\frac{2410}{577}$	$\frac{2357}{748}$	$-\frac{1133}{487}$	$\frac{2701}{328}$
$\frac{31}{50}$	$\frac{5012029}{34652500}$	$-\frac{654441}{2922500}$	$\frac{174375}{388108}$	$\frac{1}{4}$	0	0	0	$\frac{2683}{129}$	$\frac{3639}{494}$	$\frac{3745}{612}$	$-\frac{4289}{626}$	$\frac{5597}{229}$
$\frac{17}{20}$	$\frac{15267082809}{155376265600}$	$-\frac{71443401}{120774400}$	$\frac{730878875}{902184768}$	$\frac{2285395}{8070912}$	$\frac{1}{4}$	0	0	$-\frac{995}{196}$	$-\frac{3040}{1441}$	$-\frac{1076}{829}$	$\frac{449}{274}$	$-\frac{3194}{371}$
1	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$-\frac{2260}{8211}$	$\frac{1}{4}$	0	$\frac{2395}{2004}$	$\frac{761}{1482}$	$\frac{206}{569}$	$\frac{470}{1237}$	$\frac{2577}{757}$
	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$-\frac{2260}{8211}$	$\frac{1}{4}$						

Table 9: Results for the steady-state Dirichlet boundary conditions.

	I	$v(x)=3$		$v(x)=20$	
		E_∞	O_∞	E_∞	O_∞
	80	5.6E−08	—	7.7E−06	—
	160	3.7E−09	3.9	7.3E−07	3.4
$\mathbb{P}_3(4)$	320	2.4E−10	4.0	5.7E−08	3.7
	640	1.5E−11	4.0	3.9E−09	3.8
	40	3.5E−09	—	9.4E−06	—
	80	6.3E−11	5.8	3.7E−07	4.7
$\mathbb{P}_5(6)$	160	1.1E−12	5.9	9.0E−09	5.3
	320	1.7E−14	6.0	1.8E−10	5.7

4.1. Steady-state cases

To perform the numerical simulations, we consider uniform meshes of domain $\Omega = (0,1)$ constituted of I cells. Since we shall only deal with regular solutions, we use the L^∞ norm to compute the error between the solution and the approximation, which is given by

$$E_\infty(I) = \max_{i=1}^{I} |\phi_i - \bar{\phi}_i|,$$

and the convergence order between two meshes characterized by I_1 and I_2 cells is given by

$$O_\infty(I_1, I_2) = \frac{|\log(E_\infty(I_1)/E_\infty(I_2))|}{|\log(I_1/I_2)|}.$$

4.1.1. Dirichlet condition

We consider the convection-diffusion problem with homogeneous Dirichlet conditions and set $f(x) = 1$ such that the solution is given by $\phi(x) = \frac{1}{v}\left(x - \frac{\exp(vx)-1}{\exp(v)-1}\right)$. We set $\kappa(x) = 1$ and test two velocities: $v(x) = 3$ and $v(x) = 20$. We report in Table 9 the errors.The numerical scheme is stable even with a large Péclet number and provides the theoretical order (second-, fourth-, and sixth-order respectively) for the $\mathbb{P}_1$, $\mathbb{P}_3$, and $\mathbb{P}_5$ reconstructions respectively. For the large Péclet number, the convergence is effective when the mesh parameter is small enough to capture the boundary layer.

4.1.2. Total and partial Neumann conditions

Two types of Neumann condition can be prescribed whether we consider the total flux or the diffusive flux leading to two different discretizations. We study here the impact of the Neumann condition as a function of the Péclet number. To this end, we consider the convection-diffusion problem with a left boundary Dirichlet condition and a right boundary Neumann condition. For the sake of simplicity, we set $\kappa(x) = 1$ and prescribe several constant velocities for v, namely -20, -3, 3, and 20. We take $\phi(x) = \frac{1}{1+x}$ so that

Table 10: Results for the steady-state Dirichlet + total Neunann boundary conditions.

	I	$v(x) = -20$		$v(x) = -3$		$v(x) = 3$		$v(x) = 20$	
		E_∞	O_∞	E_∞	O_∞	E_∞	O_∞	E_∞	O_∞
	40	1.9E−07	—	6.3E−07	—	1.1E−06	—	8.9E+01	—
	80	2.8E−08	2.7	4.6E−08	3.8	8.3E−08	3.8	5.4E+00	4.1
$\mathbb{P}_3(4)$	160	2.5E−09	3.5	3.1E−09	3.9	5.6E−09	3.9	3.5E−01	3.9
	320	1.8E−10	3.8	2.0E−10	3.9	3.8E−10	3.9	2.3E−02	3.9
	20	1.1E−07	—	3.0E−07	—	6.8E−06	—	5.4E+01	—
	40	4.8E−09	4.5	7.7E−09	5.3	1.6E−07	5.4	4.3E+00	3.6
$\mathbb{P}_5(6)$	80	1.3E−10	5.2	1.6E−10	5.6	3.0E−09	5.7	8.1E−02	5.7
	160	2.5E−12	5.7	2.8E−12	5.8	5.2E−11	5.8	1.3E−03	5.9

the source term is $f(x) = -\frac{2}{(x+1)^3} - \frac{v}{(x+1)^2}$ and the Dirichlet conditions at point $x_{\mathrm{L}} = 0$ writes $\phi_{\mathrm{D}}(0) = 1$.

For a left total Neumann boundary condition, we have $\phi_{\mathrm{T}}(1) = \frac{1+2v}{4}$. From Table 10, where we report the L^∞ error for the four velocities, we observe that the total flux condition provides lower accurate approximations for positive velocity although we obtain the optimal convergence order for all velocities. Moreover, the accuracy degradation increases with the Péclet number and decreases with finer mesh. Discrepancies with positive velocities derive from the incompatibility to prescribe the flux for a pure convection problem. Indeed, outflow does not require external information for pure convection problem while the total flux with $\kappa(x) = 0$ implicitly imposes an external condition for the convective contribution. For $\kappa(x) > 0$ diffusion helps to incorporate the external information but a stability condition involving the Péclet number and the mesh parameter such as $\kappa h < C|v|$, $C \in \mathbb{R}$, is required. No instability appears with negative velocities since we now deal with an inflow condition for the pure convection case and one has to prescribe the external condition to fix the inflow.

For a left partial Neumann boundary condition, we have $\phi_{\mathrm{P}}(1) = \frac{1}{4}$. From Table 11, where we report the L^∞ error for the four velocities, we observe that we obtain opposite results in the sense that the scheme is stable for positive velocities while lower accuracy approximations are obtained with negative velocities. The explanation is the following. For positive valued velocity, the discretization of the convective flux turns out to be an upwind scheme which provides a very stable solution even for the diffusion-free problem ($\kappa = 0$). For negative values, the discretization turns into a downwind scheme which provides larger errors. For pure convection problem, the external information should not be prescribed and leads to an under-determined problem. When $\kappa > 0$, the diffusion scheme brings stability conditioned by the Péclet number and the mesh parameter like $ah < C|v|$, $C \in \mathbb{R}$.

To sum up, one has to prescribe the total flux Neumann condition for inflow boundary and the diffusive flux (or partial flux) Neumann condition for the outflow boundary. Notice that total flux and diffusive flux are equal when dealing with a boundary with a

Table 11: Results for the steady-state Dirichlet + partial Neunann boundary conditions.

	I	$v(x)=-20$		$v(x)=-3$		$v(x)=3$		$v(x)=20$	
		E_∞	O_∞	E_∞	O_∞	E_∞	O_∞	E_∞	O_∞
	40	1.2E+02	—	4.0E−06	—	7.3E−07	—	7.8E−07	—
	80	5.7E+00	4.4	2.4E−07	4.1	4.9E−08	3.9	5.1E−08	3.9
$\mathbb{P}_3(4)$	160	3.3E−01	4.1	1.5E−08	4.0	3.2E−09	3.9	3.3E−09	4.0
	320	2.0E−02	4.0	9.1E−10	4.0	2.1E−10	4.0	2.1E−10	4.0
	20	2.5E+00	—	1.1E−06	—	4.2E−07	—	5.9E−07	—
	40	5.4E−01	2.2	8.0E−09	7.1	8.9E−09	5.6	1.2E−08	5.6
$\mathbb{P}_5(6)$	80	6.4E−03	6.4	1.5E−10	5.7	1.7E−10	5.7	2.1E−10	5.9
	160	8.9E−05	6.2	2.7E−12	5.8	2.9E−12	5.9	3.3E−12	6.0

null velocity.

4.2. Time-dependent cases

To perform the numerical simulations, we consider uniform meshes for domain $\Omega = (0,1)$ constituted of I cells. The computation are carried out up to the final time $t_\mathrm{f} = 1$ for the explicit methods and $t_\mathrm{f} = 10$ for non-explicit methods. Several time schemes will be tested where we identify by "`BT`" the *Butcher Tableaux* and by "`EBT`" the *Extended Butcher Tableaux*. Errors between the solution and the approximation are evaluated at the final time with

$$E_\infty(I) = \max_{i=1}^{I} |\phi_i^N - \bar{\phi}_i^N|.$$

4.2.1. An explicit sixth-order scheme

We aim at comparing the `BT` and `EBT` methods when dealing with time-independent or time-dependent Dirichlet conditions. For this purpose, we consider the classical explicit RK3 method for a pure diffusive problem taking $\kappa(x,t) = 1$ and $v(x,t) = 0$ given by Table 2 and by Table 5, respectively. In a first case we assume that the exact solution writes $\phi(x,t) = (x-1)\ln(x+1)\exp(-t)$ and deduce the source term $f(x,t) = \left[(1-x)\ln(x+1) - \frac{x+3}{(x+1)^2}\right]\exp(-t)$ with time-independent Dirichlet condition. In a second case, we assume that the exact solution is $\phi(x,t) = \cosh(x)\exp(-t)$ such that the source term is $f(x,t) = -2\cosh(x)\exp(-t)$ and the Dirichlet conditions are now time-dependent. Since we use a $\mathbb{P}_5$ reconstruction and one has to respect stability condition for the explicit time scheme, the time step is given by $\Delta t = \frac{1}{3}h^2$ to provide a full sixth-order scheme with respect to h. We reproduce the errors and convergence rates with the `BT` and `EBT` methods for the first situation in Table 12 and in Table 13 for the second situation . We observe that the effective order are optimal for both methods in the time-independent case whereas for the time-dependent with `BT` there is a clear reduction in the accuracy to a fourth-order method. The `EBT` method manages to recover the optimal order and shows the effectiveness of the correction.

Table 12: Results for the time-independent Dirichlet example with $\kappa(x,t) = 1$, $v(x,t) = 0$, and $\phi(x,t) = (x-1)\ln(x+1)\exp(-t)$ for RK3

	I	BT		EBT	
		E_∞	O_∞	E_∞	O_∞
	10	1.6E−06	—	1.6E−06	—
$\mathbb{P}_5(6)$	20	5.0E−08	5.0	5.0E−08	5.0
$\Delta t = \frac{1}{3}h^2$	40	1.2E−09	5.4	1.2E−09	5.4
	80	2.2E−11	5.7	2.2E−11	5.7

Table 13: Results for the time-dependent Dirichlet example with $\kappa(x,t) = 1$, $v(x,t) = 0$, and $\phi(x,t) = \cosh(x)\exp(-t)$ for RK3

	I	BT		EBT	
		E_∞	O_∞	E_∞	O_∞
	10	5.2E−07	—	3.7E−08	—
$\mathbb{P}_5(6)$	20	3.2E−08	4.0	6.1E−10	5.9
$\Delta t = \frac{1}{3}h^2$	40	2.0E−09	4.0	1.0E−11	5.9
	80	1.3E−10	4.0	1.6E−13	6.0

Table 14: Orders of convergence for the time-dependent Dirichlet problems with $\kappa(x,t) = 1$, $v(x,t) = 0$, $\phi(x,t) = x^\sigma t^\tau$, polynomial reconstructions of degree 5, and time step $\Delta t = \frac{1}{3}h^2$ for RK3 BT/EBT (expected order of convergence: 6)

σ \ τ	0	1	2	3	4
0	E/E	E/E	4/E	4/6	4/6
1	E/E	E/E	4/E	4/6	4/6
2	E/E	E/E	4/6	4/6	4/6
3	E/E	E/E	4/6	4/6	4/6
4	E/E	E/E	4/6	4/6	4/6

Table 15: Results for the time-independent Dirichlet example with $\kappa(x,t) = 1$, $v(x,t) = 0$, and $\phi(x,t) = (x-1)\ln(x+1)\exp(-t)$ for RK4

	I	BT		EBT	
		E_∞	O_∞	E_∞	O_∞
	10	3.4E−07	—	3.4E−07	—
$\mathbb{P}_7(8)$	20	4.0E−09	6.4	4.0E−09	6.4
$\Delta t = \frac{1}{3}h^2$	40	2.9E−11	7.1	2.9E−11	7.1
	80	1.6E−13	7.5	1.6E−13	7.5

To go deeper in the convergence study, we shall check the convergence order for monomial functions of the form $x^\sigma t^\tau$ with $\sigma, \tau \in \{0,1,2,3,4\}$. We prescribe the corresponding source term and Dirichlet condition and carry out the computation using the $\mathbb{P}_5$ reconstruction in space. We sum-up in Table 14 the test convergence where "E" (for Exact) means that the approximation corresponds to the exact solution otherwise we indicate the rate of convergence. Moreover in expression 4/6 the first digit indicates the convergence order with the BT while the second digit corresponds to the EBT one. Relevant cases are the monomial functions t^2 and xt^2 which are exactly solved using the EBT scheme while the traditional RK3 only provides a fourth-order approximation. In the same way, for monomial functions such as x^2t^2 or xt^3, the EBT technique gives a sixth-order approximation whereas the BT one only reaches the fourth-order.

4.2.2. An explicit eighth-order scheme

We consider the diffusion case where we set $\kappa(x,t) = 1$ and $v(x,t) = 0$ and we use a $\mathbb{P}_7$ reconstruction in space associated to the explicit RK4 Runge-Kutta method proposed in Table 6. As in the previous case, the time-independent and time-dependent Dirichlet conditions are addressed to assess the EBT method capacity to improve convergence order. For stability reasons, we impose $\Delta t = \frac{1}{3}h^2$ and expect a global convergence of order 8. In Table 15 we present the errors and convergence rates for the first situation and in Table 16 we present the errors and convergence rates for the second situation.

As in the previous case, the EBT technique clearly improves the accuracy of the

Table 16: Results for the time-dependent Dirichlet example with $\kappa(x,t) = 1$, $v(x,t) = 0$, and $\phi(x,t) = \cosh(x)\exp(-t)$ for RK4

	I	BT		EBT	
		E_∞	O_∞	E_∞	O_∞
	10	2.8E−06	—	3.5E−09	—
$\mathbb{P}_7(8)$	20	1.6E−07	4.1	5.2E−11	6.1
$\Delta t = \frac{1}{3}h^2$	40	1.0E−08	4.0	8.2E−13	6.0
	80	6.4E−10	4.0	1.1E−14	6.3

Table 17: Orders of convergence for the time-dependent Dirichlet problems with $\kappa(x,t) = 1$, $v(x,t) = 0$, $\phi(x,t) = x^\sigma t^\tau$, polynomial reconstructions of degree 7, and time step $\Delta t = \frac{1}{3}h^2$ for RK4 BT/EBT (expected order of convergence: 8)

σ \ τ	0	1	2	3	4
0	E/E	E/E	4/E	4/E	4/8
1	E/E	E/E	4/E	4/E	4/8
2	E/E	E/E	4/6	4/6	4/6
3	E/E	E/E	4/6	4/6	4/6
4	E/E	E/E	4/6	4/6	4/6

method. For the time-independent case, we get the optimal order for both cases whereas the classical Runge-Kutta fails to accurately integrate the time-dependent boundary conditions and dramatically cuts the accuracy by two orders of magnitude. As in the previous section, we analyse in Table 17 the convergence of the method for the monomials $x^\sigma t^\tau$ considering polynomial reconstructions of degree 7 and the RK4 method with time step $\Delta t = \frac{1}{3}h^2$. We observe that we get the exact solution for monomials t^2 and t^3 by construction and achieve the eighth-order of convergence for t^4 whereas the BT method only provides the fourth-order convergence. Unfortunately, for some monomials functions like x^2t^2, we only achieve a sixth-order of convergence since solving exactly t^2 and t^3 do not guarantee solving exactly x^2t^2. It results that polynomial Taylor expansions will be approximated with a sixth-order method which explains the sixth-order convergence observed with function $\cosh(x)\exp(-t)$ as shown in Table 16.

4.2.3. An implicit fourth-order scheme

We aim to test an implicit version of the Runge-Kutta method in order to use an unconditionally stable scheme allowing larger time step. In section 3.7 we have proposed two versions for the EBT technique applied to the ESDIRK4 method, namely (LS) and (AC), since the number of stages is larger than the number of conditions (Tables 7 and 8, respectively). To perform the simulation, we consider a pure diffusive problem setting $\kappa(x,t) = 1$ and $v(x,t) = 0$ and consider again time-independent and time-dependent Dirichlet boundary conditions. The exact solution writes $\phi(x,t) = (x-1)\ln(x+1)\exp(-t)$ for

Table 18: Results for the time-independent Dirichlet example with $\kappa(x,t) = 1$, $v(x,t) = 0$, and $\phi(x,t) = (x-1)\ln(x+1)\exp(-t)$ for ESDIRK4

	I	BT		EBT (LS-way)		EBT (AC-way)	
		E_∞	O_∞	E_∞	O_∞	E_∞	O_∞
	20	2.5E−09	—	2.5E−09	—	2.5E−09	—
$\mathbb{P}_3(4)$	40	1.6E−10	3.9	1.6E−10	3.9	1.6E−10	3.9
$\Delta t = 10h$	80	1.1E−11	4.0	1.1E−11	4.0	1.1E−11	4.0
	160	6.8E−13	4.0	6.8E−13	4.0	6.8E−13	4.0

Table 19: Results for the time-dependent Dirichlet example with $\kappa(x,t) = 1$, $v(x,t) = 0$, and $\phi(x,t) = \cosh(x)\exp(-t)$ for ESDIRK4

	I	BT		EBT (LS-way)		EBT (AC-way)	
		E_∞	O_∞	E_∞	O_∞	E_∞	O_∞
	20	1.3E−04	—	2.0E−05	—	2.2E−05	—
$\mathbb{P}_3(4)$	40	1.1E−05	3.6	1.1E−06	4.2	1.1E−06	4.3
$\Delta t = 10h$	80	1.2E−06	3.2	6.6E−08	4.0	6.7E−08	4.1
	160	1.3E−07	3.1	4.2E−09	4.0	4.0E−09	4.1
	10	4.3E−08	—	3.6E−08	—	3.5E−08	—
$\mathbb{P}_5(6)$	20	1.1E−09	5.4	6.0E−10	5.9	5.9E−10	5.9
$\Delta t = h^{1.5}$	40	3.4E−11	4.9	9.8E−12	5.9	9.7E−12	5.9
	80	1.4E−12	4.6	2.0E−13	5.6	2.0E−13	5.6

the first situation while function $\phi = \cosh(x)\exp(-t)$ corresponds to the second one. Due to unconditional stability, we take a larger time step $\Delta t = 10h$ and apply a $\mathbb{P}_3$ polynomial reconstruction to achieve a global fourth-order accurate scheme. An other possibility consists in choosing $\Delta t = h^{1.5}$ with a $\mathbb{P}_5$ reconstruction in space to provide a global sixth-order accurate scheme. The error and convergence orders are reproduced in Tables 18 and 19 for the time-independent and time-dependent Dirichlet boundary conditions, respectively. As expected, both methods provide the optimal order for the time-independent case. For the time-dependent boundary condition the BT method only provides a third-order scheme wiht the $\mathbb{P}_3$ reconstruction and we observe an order 4.5 with $\Delta = h^{1.5}$ and a $\mathbb{P}_5$ reconstruction. On the contrary, EBT method clearly releases the accuracy reduction and provides the optimal order for the two situations. We highlight that both methods — LS and AC —, give very similar errors.

4.2.4. An explicit sixth-order scheme with Neumann condition

We now test the explicit RK3 schemes in time for a pure diffusive problem with $\kappa(x,t) = 1$ and Neumann conditions. The exact function is $\phi(x,t) = \cosh(x)\exp(-t)$

Table 20: Results for the time-dependent Neumann example with $\kappa(x,t) = 1$, $v(x,t) = 0$, and $\phi(x,t) = \cosh(x)\exp(-t)$ for RK3

	I	BT		EBT	
		E_∞	O_∞	E_∞	O_∞
	10	7.1E−09	—	3.5E−10	—
$\mathbb{P}_7(8)$	14	1.3E−09	5.0	3.1E−11	7.2
$\Delta t = \frac{1}{3}h^2$	17	5.1E−10	5.0	8.1E−12	6.9
	20	2.3E−10	5.0	2.9E−12	6.3

which provides the function $\phi_{\mathrm{T}}(x,t) = -\sinh(x)\exp(-t)$. To perform the simulations, we take $\Delta t = \frac{1}{3}h^2$ for the sake of stability and use the $\mathbb{P}_7$ to better highlight the convergence in time. Indeed, the approximation in space is more accurate hence the errors derives from the time discretization. We presents the errors and convergence rates in Table 20. We first observe that we get a fifth-order converge for the BT technique in place of the fourth-order obtained in the case of time-dependent Dirichlet condition. The observation is confirmed with the EBT technique: for convection-diffusion problem with Neumann conditions, we obtain a better approximation (one order of magnitude) than for the case with Dirichlet condition. On the other hand, the scheme in time based on Extended Butcher Tableau provides an optimal seventh-order.

4.2.5. A complex example

To end the numerical section we compare the three discretizations in time equipped with the BT or the EBT method for a convection-diffusion problem with time and space dependent coefficients. We take $\kappa(x,t) = 3e^{-t}\cosh(x)$ and $v(x,t) = -0.5e^{t}(x^2+1)$ and assume that the exact function writes $\phi(x,t) = \cosh(x)\exp(t)$ which implies that $f(x,t) = \exp(t)\cosh(x) - 3\cosh^2(x) - 3\sinh^2(x) - \frac{1}{2}\sinh(x)(x^2+1) - x\cosh(x)$. We prescribe the Dirichlet condition both at $x = 0$ and $x = 1$. The explicit RK3 scheme is used with $\Delta t = \frac{1}{10}h^2$ and the $\mathbb{P}_5$ reconstruction in space to provide a global sixth-order scheme. The explicit RK4 scheme is using the same time step and a $\mathbb{P}_7$ reconstruction in space to provide a global eighth-order scheme. For the implicit ESDIRK4, we use larger time step $\Delta t = 10h$ and apply a $\mathbb{P}_3$ reconstruction in space to provide a global fourth-order scheme. Table 21 clearly shows the effectiveness of the EBT method since we recover the optimal sixth-order method even with non-constant diffusion and velocity.

The situation of the RK4 scheme presented in Table 22 highlights the dramatic influence of the boundary condition with very high-order approximation. The BT method only produces a global fourth-order scheme whereas the extended version recover the optimal order.

In the implicit case presented in Table 23, the EBT method provides the optimal order in contrast with the BT method which strongly cuts by an half the accuracy order.

Table 21: Results for the time-dependent Dirichlet example with $\kappa(x,t) = 3\cosh(x)\exp(-t)$, $v(x,t) = -\frac{1}{2}\exp(t)(1+x^2)$, and $\phi(x,t) = \cosh(x)\exp(t)$ for RK3

	I	BT		EBT	
		E_∞	O_∞	E_∞	O_∞
	10	2.7E−07	—	2.5E−07	—
$\mathbb{P}_5(6)$	20	1.2E−08	4.5	4.3E−09	5.9
$\Delta t = \frac{1}{10}h^2$	30	2.2E−09	4.1	3.9E−10	5.9
	40	7.0E−10	4.0	7.1E−11	5.9

Table 22: Results for the time-dependent Dirichlet example with $\kappa(x,t) = 3\cosh(x)\exp(-t)$, $v(x,t) = -\frac{1}{2}\exp(t)(1+x^2)$, and $\phi(x,t) = \cosh(x)\exp(t)$ for RK4

	I	BT		EBT	
		E_∞	O_∞	E_∞	O_∞
	10	1.5E−08	—	3.2E−09	—
$\mathbb{P}_7(8)$	20	8.8E−10	4.1	1.4E−11	7.8
$\Delta t = \frac{1}{15}h^2$	30	1.7E−10	4.0	5.8E−13	7.9
	40	5.5E−11	4.0	5.6E−14	8.1

Table 23: Results for the time-dependent Dirichlet example with $\kappa(x,t) = 3\cosh(x)\exp(-t)$, $v(x,t) = -\frac{1}{2}\exp(t)(1+x^2)$, and $\phi(x,t) = \cosh(x)\exp(t)$ for ESDIRK4

	I	BT		EBT	
		E_∞	O_∞	E_∞	O_∞
	40	4.6E−02	—	7.2E−02	—
$\mathbb{P}_3(4)$	80	9.2E−03	2.3	5.1E−03	3.8
$\Delta t = 10h$	160	2.5E−03	1.9	3.5E−04	3.9
	320	5.6E−04	2.1	2.3E−05	3.9

5. Conclusion

We have shown that straightforward application of the Runge-Kutta time schemes based on the Butcher Tableau fails to provide the optimal convergence order in numerous cases. The main reason is a wrong synchronization between the Runge-Kutta scheme applied to the linear system deriving from the finite volume formulation and the boundary conditions. Constant in space solution are not preserved and numerical perturbations are generated at the boundary and further propagate into the whole domain. We have proposed a strategy to cure the undesirable phenomenon introducing an extension to the Butcher Tableau such that we preserve solutions t^τ for Dirichlet boundary conditions and xt^τ for Neumann boundary conditions up to a certain degree. Numerical simulations show the high effectiveness of the method for both boundary condition types. Extension of the method for the two- and three-dimension geometries seems straightforward and will be investigated in a future study.

Acknowledgements

The authors would like to thank Dr. Raphael Loubère for his suggestions. This research was financed by FEDER Funds through Programa Operacional Factores de Competitividade — COMPETE and by Portuguese Funds through FCT — Fundação para a Ciência e a Tecnologia, within the Projects PEst-C/MAT/UI0013/2011 and PTDC/MAT/121185/2010 and also by the Project *Programa Pessoa Cooperação Transnacional Portugal/França Fundação para a Ciência e a Tecnologia* under the reference 130631467126259.

6. Bibliography

[1] L. Ivan, C.P.T. Groth, High-Order Solution-Adaptive Central Essentially Non-Oscillatory (CENO) Method for Viscous Flows AIAA 2011-367 (2011) 1–30.

[2] C. Ollivier-Gooch, M. Van Altena, A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation, Journal of Computational Physics Archive 181:2 (2002), 729–752.

[3] R. Eymard, T. Gallouët, R. Herbin, The finite volume method, Handbook for Numerical Analysis, Ph. Ciarlet J.L. Lions eds., North Holland, (2000), 715–1022.

[4] S. Clain, G.J. Machado, J.M. Nobrega, R.M.S. Pereira, A sixth-order finite volume method for multidomain convection-diffusion problem with discontinuous coefficients, Computer Methods in Applied Mechanics and Engineering, 267 (2013), 43–64.

[5] S. Clain, S. Diot, R. Loubère, A high-order polynomial finite volume method for hyperbolic system of conservation laws with Multi-dimensional Optimal Order Detection (MOOD), Journal of Computational Physics, 230 (2011), 4028–4050.

[6] S. Diot, S. Clain, R. Loubère, Multi-dimensional Optimal Order Detection (MOOD) — A very high-order Finite Volume Scheme for conservation laws on unstructured meshes, Proceeding for the sixth Finite Volume and Complex Application, Springer Verlag Editor, 1 (2011), 263–271.

[7] J.A. Hernández, High-order finite volume schemes for the advection-diffusion equation, International Journal for Numerical Methods in Engineering, 53 (2002), 1211–1234.

[8] E.F. Toro, A. Hidalgo, ADER finite volume schemes for nonlinear reaction-diffusion equations Applied Numerical Mathematics, 59:1 (2009), 73–100.

[9] J.C. Butcher, Numerical methods for Ordinary Differential Equations, John Wiley & Sons, (2008)

[10] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong Stability Preserving high-order time discretization methods, ICASE report 2000-15 (2000) 1–23.

[11] M. Dumbser, O. Zanotti, A. Hidalgo, D. Balsara, ADER-WENO finite volume schemes with space-time adaptive mesh refinement, J. Comput. Physics 248 (2013), 257–286.

[12] A. Hidalgo, M. Dumbser, ADER Schemes for Nonlinear Systems of Stiff Advection Diffusion Reaction Equations J. Sci. Comput. 48 (2011) 173–189.
[13] E. Bertolazzi, G. Manzini, A unified treatment of boundary conditions in least-square based finite-volume methods, Comp. & Math. with applications 49 (2005) 1755–1765.
[14] P. Chandrashekar, A. Garg, Vertex-centroid finite volume scheme on tetrahedral grids for conservation laws, Comp. & Math. with applications 65 (2013) 58-74.