

Bawfeh Kingsley Kometa

Semi-Lagrangian methods and new integration schemes for convection-dominated problems

Thesis for the degree of Philosophiae Doctor

Trondheim, November 2011

Norwegian University of Science and Technology
Faculty of Information Technology, Mathematics and
Electrical Engineering
Department of Mathematical Sciences



NTNU – Trondheim
Norwegian University of
Science and Technology

NTNU

Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Information Technology, Mathematics and Electrical Engineering
Department of Mathematical Sciences

© Bawfeh Kingsley Kometa

ISBN 978-82-471-3107-7 (printed ver.)
ISBN 978-82-471-3108-4 (electronic ver.)
ISSN 1503-8181

Doctoral theses at NTNU, 2011:270

Printed by NTNU-trykk

To my dear mum

Netenu Alice Ndula

Preface

This thesis is in partial fulfillment of the requirements for the degree of *Philosophiae Doctor* (PhD) at the Norwegian University of Science and Technology (NTNU). It contains results from scientific research work carried out in the field of Numerical and Applied Mathematics. The work was fully funded by the Department of Mathematical Sciences (IMF) at NTNU, over a duration of four years running from August 2007 to August 2011. The main place of work has been IMF.

All numerical results presented were carried out using the scientific computing software MATLAB. Other useful resources exploited include the online database sources such as MathSciNet, ScienceDirect, Google, BIBSYS among others. All documentation of the work was done using LaTeX.

Structure of thesis

This thesis consists of an introduction and five papers, arranged in a total of six chapters. A brief summary of the results in each of the papers is given in Section 1.4 of the introduction.

All five articles involve analysis, implementations and programming aspects. My contribution to these articles has been substantial in all respects.

Bawfeh Kingsley Kometa,
Trondheim,
November 2011.

Acknowledgements

I would like to acknowledge my PhD supervisor, Professor Elena Celledoni, for her tireless commitment and follow-up to see this work come through. I cannot forget her single-hearted support and encouragement right from the inception of my PhD career. I also want to appreciate the IMF department for funding this work. Special regards to Olivier Verdier for vital corporations. To Tormod Bjøntegaard for a couple of useful discussions. To the entire staff and members of the Numerical Analysis group at IMF for creating an exciting working environment. To my Norwegian family friends Olaf Normann and Erling Fossan for making my stay here in Trondheim, Norway, worth remembering. To several colleagues in and around Norway who contributed via fruitful discussions and talks at conferences/seminars such as *Manifold and Geometric Integration Colloquium* (MaGIC), *International Conference on Spectral and High Order Methods* (ICOSAHOM), *International Conference on Numerical and Applied Mathematics* (ICNAAM) etc. To a large host of friends (not mentioned here) both within and without NTNU, in church and out of church, within and outside Norway, whose encouragements in one way or the other boosted my confidence. To my mum for every moral support chipped in along the way. Finally, to the Most High God for always being there both in the good and the bad times.

Bawfeh Kingsley Kometa,

Trondheim,
November 2011.

Table of Contents

1	Introduction	1
1.1	Background	1
1.2	Scope of thesis	2
1.3	Key concepts	3
1.3.1	Semi-Lagrangian methods	3
1.3.2	Convection-dominated problems	5
1.3.3	Implicit-explicit methods	5
1.3.4	Commutator-free Lie-group exponential integrators	5
1.3.5	Krylov subspace methods and matrix exponentials	8
1.3.6	Notations	8
1.4	Summary of results	9
	References	11
2	Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems	19
2.1	Introduction	19
2.2	Presentation of the new class of methods	22
2.2.1	Stability function	24
2.2.2	Order conditions	25
2.2.3	Methods of order 2 and 3	33
2.3	Numerical tests	36
2.3.1	Exponentials	37
2.3.2	Korteweg de Vries equation (KdV)	39
2.3.3	1D Viscous Burgers' equation	40
2.3.4	Linear convection and convection-diffusion in 2D	42
2.4	Appendix	44
2.4.1	Commutator-free methods for integration of ODEs on manifolds	44
	References	45

Table of Contents

3	Order conditions for the semi-Lagrangian exponential integrators	51
3.1	Introduction	51
3.2	Deriving the order conditions	52
3.3	Order conditions for orders 1 – 3	55
3.4	Extra coupling conditions for order 4	61
	References	65
4	Semi-Lagrangian multistep exponential integrators for index 2 DAEs	69
4.1	Introduction	69
4.1.1	The BDF-CF methods	72
4.2	Construction of the methods	73
4.2.1	Second order method (BDF2-CF)	74
4.2.2	Third order method (BDF3-CF)	76
4.2.3	Fourth order method (BDF4-CF)	77
4.3	Some convergence results	77
4.3.1	Local error	80
4.3.2	Global Error	80
4.3.3	Numerical example	80
4.3.4	Numerical test on Navier-Stokes	81
4.4	Stability of the BDF-CF methods	84
4.4.1	A nonlinear problem	84
4.4.2	Linear Stability	85
4.5	Appendix	89
4.5.1	Definition of norms	89
4.5.2	Order conditions for order 3 method BDF3-CF	89
4.5.3	DIRK-CF methods and commutator-free methods	90
	References	92
5	On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems	99
5.1	Introduction	99
5.2	DG methods for hyperbolic conservation laws	101
5.2.1	Weak formulation on a broken Sobolev space	101
5.2.2	The discrete weak formulation on a piece-wise polynomial space	102
5.3	Semi-Lagrangian DG schemes and Commutator-free exponential integrators	103
5.3.1	The semi-Lagrangian discontinuous Galerkin (SLDG) methods	103
5.3.2	Commutator-free Lie group method	104
5.4	Numerical results	105
5.4.1	Pure advection in 1D	105
5.4.2	Pure advection in 2D	106
	References	110

6	Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations	115
6.1	Introduction	115
6.2	Projection methods for the incompressible Navier-Stokes equations	117
6.2.1	Leray projector	117
6.2.2	Incompressible Navier-Stokes and projections	118
6.3	High order implicit-explicit and semi-Lagrangian methods of Runge-Kutta type	120
6.3.1	IMEX Runge-Kutta	120
6.3.2	Semi-Lagrangian IMEX Runge-Kutta	120
6.4	Numerical experiments	121
6.4.1	Implementation issues	122
6.4.2	Temporal order tests for the IMEX methods	123
6.4.3	Temporal order tests for the DIRK-CF methods	125
6.4.4	Lid-driven cavity flow in 2D	126
6.4.5	Shear layer roll-up problem	128
	References	132

Chapter 1

Introduction

1.1 Background

The model problems considered in this thesis are linear and nonlinear convection-diffusion problems with a dominating convection term. These models are challenging and ubiquitous in applications, an example being the numerical simulation of internal waves phenomena occurring between the layers of a stratified flow. In Norwegian fjords, for example, layers of stratified flow with different temperature and salt concentrations occur due to ice melting and freshwater supply from rivers. Internal waves are caused by tides and could have a dramatic influence on the ecosystem.

We hereby consider convection-diffusion PDE models, depending on a viscosity parameter ν , of the type

$$\frac{\partial}{\partial t} \mathbf{u}(\mathbf{x}, t) + \mathbf{V} \cdot \nabla \mathbf{u}(\mathbf{x}, t) = \nu \nabla^2 \mathbf{u}(\mathbf{x}, t) + \mathbf{f}(\mathbf{x}), \quad (1.1.1)$$

where $\mathbf{x} \in \Omega \subset \mathbb{R}^d$, $t \in (0, T]$ such that $\mathbf{u} : \Omega \times (0, T] \rightarrow \mathbb{R}^d$, $\mathbf{V} : \Omega \times (0, T] \rightarrow \mathbb{R}^d$, and $\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x})$. In the nonlinear case the velocity field \mathbf{V} also depends on the unknown \mathbf{u} . The case when the parameter ν tends to zero is particularly interesting and very challenging from a numerical point of view. In this case the numerical discretizations often lead to the phenomenon of numerical dispersion. In the 4th and 6th chapters we adapt and apply the methods to the Navier-Stokes equations.

In this work, a new class of time integration methods which we believe has a large potential for convection-dominated problems has been studied. These methods, earlier presented in [6], are particular in that while the diffusion part is integrated by an implicit scheme, the flow of the linearized convection term is computed accurately. For this reason we call them exponential integrators. The class of time integration methods we present and study in this thesis are designed to attain good performance in convection-dominated problems. The accurate approximation of pure convection flows appears as a building

Chapter 1. Introduction

block for these methods. The methods are obtained by composition of exact convection flows and the implicit time integration of the diffusion, the overall method takes the form of an implicit-explicit exponential integrator [36, 24].

A simple example is the following transport-diffusion algorithm studied by Pironneau in [44],

$$\begin{aligned} \frac{D\tilde{\mathbf{u}}}{Dt} &= 0, \quad \tilde{\mathbf{u}}_{n+1}(\mathbf{x}) = \mathbf{u}_n(\mathbf{x}), \\ \mathbf{u}_{n+\frac{1}{2}}(\mathbf{x}) &= \tilde{\mathbf{u}}_n(\mathbf{x}), \\ \mathbf{u}_{n+1} &= \mathbf{u}_{n+\frac{1}{2}} + h\nu\nabla^2\mathbf{u}_{n+1} + h\mathbf{f}, \end{aligned} \quad (1.1.2)$$

where h is the time step, while $\mathbf{u}_n, \mathbf{u}_{n+1}$ denote the numerical solution at time levels t_n, t_n+h respectively, and $\mathbf{u}_{n+\frac{1}{2}}$ is an intermediate value. The exact integration of the linear pure convection problem

$$\frac{D\tilde{\mathbf{u}}}{Dt} := \frac{\partial}{\partial t}\tilde{\mathbf{u}} + \mathbf{u}_n \cdot \nabla\tilde{\mathbf{u}} = 0, \quad (1.1.3)$$

arises as a building block in the integration method, and can be achieved by computing characteristics $\chi(\tau)$,

$$\begin{aligned} \frac{d\chi}{d\tau} &= \mathbf{u}_n(\chi(\tau)), \quad \chi(t_n + h) = \mathbf{x}, \\ \tilde{\mathbf{u}}_n(\mathbf{x}) &= \mathbf{u}_n(\chi(t_n)). \end{aligned} \quad (1.1.4)$$

The resulting methods are semi-Lagrangian.

The study of higher order time integrators of this type is also motivated by some observations in recent work by Xiu and Karniadakis [54], and also Giraldo *et al.* [22]. The semi-Lagrangian methods have gained significant popularity in recent years, especially in meteorological and geophysical modeling (see for example [22, 31, 13, 4, 52] and references therein). Applications to problems of fluid dynamics were pioneered in [44] among others. Stability and error analysis have been studied in [44] for the first order method (1.1.2), and more recently by Falcone and Ferretti [19] for linear advection problems. In this work high order semi-Lagrangian methods have been studied.

Preliminary work illustrating the potential of the methods has been presented by Celledoni in [5, 6].

1.2 Scope of thesis

The aim of the thesis has been to contribute in finding robust high order time integrators for convection-dominated problems.

The authors in [54] pointed out that the use of low order semi-implicit methods in the case of direct numerical simulation of turbulent flows leads to prohibitive time step restrictions. In fact the time step dictated by the CFL condition can be several orders of magnitude smaller than the intrinsic temporal scale of the problem predicted by the theory. On the other hand semi-Lagrangian schemes allow for larger time steps (cf.

[54]) than Eulerian schemes, and show stable behaviour beyond the nominal limits set by CFL restrictions. The new class of higher order semi-Lagrangian methods studied here, take advantage of these properties. In this thesis we develop the methods originally proposed in [5] for linear convection diffusion problems and obtain methods of one-step and multi-step type for nonlinear convection diffusion equations. We show that the methods achieve high temporal order and study their performance in a variety of convection-dominated model problems. We combine semi-Lagrangian exponential integrators and discontinuous Galerkin discretizations for advection problems. We finally study suitable ways to adapt the various schemes to the case of the incompressible Navier-Stokes equations.

1.3 Key concepts

We shall describe some key concepts applicable to this thesis work.

1.3.1 Semi-Lagrangian methods

A pure linear convection of a scalar field u can be expressed in the *conservative form*

$$\frac{\partial}{\partial t}u + \nabla \cdot (\mathbf{V}u) = 0, \quad (1.3.1)$$

or in the *advective or Lagrangian form*

$$\frac{\partial}{\partial t}u + (\mathbf{V} \cdot \nabla)u = 0, \quad (1.3.2)$$

where $\mathbf{V} = \mathbf{V}(\mathbf{x}, t)$ is the advection velocity. The two forms are equivalent when \mathbf{V} is divergence-free (i.e. $\nabla \cdot \mathbf{V} = 0$). Associated to these forms are two classes of semi-Lagrangian methods popular in the literature: the conservative semi-Lagrangian methods [55, 10, 14, 17] (associated to the form (1.3.1)), and the traditional semi-Lagrangian methods [15, 53, 44] (associated to the form (1.3.2)). Common to either approach are the concepts of tracing characteristics and interpolation or reconstruction. In this work we have used the latter approach, which is briefly outlined here. Consider the characteristic paths $\chi(t)$, satisfying the equation

$$\frac{d\chi}{dt} = \mathbf{V}(\chi(t), t), \quad \chi(t_0) = \mathbf{x}_0.$$

We note that along the characteristics the equation (1.3.2) takes the total derivative form

$$\frac{D}{Dt}u(\chi(t), t) = 0$$

so that u is constant along the characteristics and

$$u(\chi(t), t) = u(\chi(t_0), t_0).$$

Chapter 1. Introduction

The traditional semi-Lagrangian routine for solving (1.3.2) over a time interval $[t_n, t_{n+1}]$ with step size $h := t_{n+1} - t_n$ involves two main steps:

Tracing characteristics

We assume that the solution at time t_n , namely u_n , is given. Particles in the flow are assumed to move along characteristic paths $\chi = \chi(t)$ such that at time t_{n+1} the particles have arrived at points \mathbf{x} belonging to a *fixed* computational grid (in the domain of the solution). But at time t_n the particles were at points $\tilde{\mathbf{x}} = \chi(t_n)$ that do not necessarily coincide with the computational grid. The points $\tilde{\mathbf{x}}$ are referred to as the *departure points* or *feet of characteristics*. We compute such points by solving the characteristic equation

$$\frac{d\chi}{d\tau} = \mathbf{V}(\chi(\tau), \tau), \text{ for } \tau \in (t_n, t_{n+1}), \text{ given } \chi(t_{n+1}) = \mathbf{x}. \quad (1.3.3)$$

This equation is solved *backward* in time using a suitable numerical method for ordinary differential equations (ODE) to obtain the departure points $\chi(t_n)$.

Interpolation

Generally the departure points are not points on the fixed grid, while we assume that the numerical solution at time level t_n is explicitly known only at grid points. Therefore $u(\tilde{\mathbf{x}})$ is recovered using a suitable interpolation procedure and we obtain the solution of (1.3.2) at time t_{n+1} as

$$u_{n+1}(\mathbf{x}) := u_n(\tilde{\mathbf{x}}).$$

Notice that (1.3.3) is naturally solved *backward* in time. However, in other formulations of (1.3.3) the feet of the characteristics are set at grid points $\chi(t_n) = \mathbf{x}$, and (1.3.3) is solved *forward* in time to obtain the *arrival points* $\chi(t_{n+1})$. Both approaches work fine, although they are not the same (see e.g. [17]). In this thesis, the characteristic equations are solved backward in time.

We emphasize that the success of the semi-Lagrangian approach depends on the accuracy of the characteristic tracing and interpolation methods used. Runge-Kutta methods of reasonable accuracy are commonly used for tracing the characteristics. More options include analytic integration, power series approximations (see for example [52] and references therein). The choice of interpolation method used is largely influenced by the kind of computational grid used for the spatial domain or the spatial approximation space used. In the literature we observe that Lagrange, cubic spline, piecewise cubic Hermite, piecewise parabolic, monotone interpolations etc [21, 46, 49, 40] are being used. Standard Lagrange interpolation methods (not suitable for uniform grids) are however sufficiently accurate for use on a pseudo-spectral or spectral element grid based on Jacobi polynomials (see e.g. [23, 54]). For an extensive review on various semi-Lagrangian methods we refer to [50, 3]. Within the finite-element framework the terms *characteristic-Galerkin* and *Lagrange-Galerkin* are commonly used in association with semi-Lagrangian methods [38, 32, 39, 45, 23].

1.3.2 Convection-dominated problems

This terminology is generally used in the literature to refer to a convection-diffusion model with relatively small diffusion parameter. Numerical methods for convection-dominated problems are very important, as stated in [11]: “*Practical problems in which convection plays an important role arise in applications as diverse as meteorology, weather-forecasting, oceanography, gas dynamics, turbomachinery, turbulent flows, granular flows, oil recovery simulation, modeling of shallow waters, transport of contaminant in porous media, visco-elastic flows, semiconductor device simulation, magneto-hydrodynamics, and electro-magnetism, among many others. This is why devising robust, accurate and efficient methods for numerically solving these problems is of considerable importance.*” For a short review on numerical methods for convection-dominated problems we refer to [18].

1.3.3 Implicit-explicit methods

The semi-discretization (in space) of several convection-diffusion models results in ordinary differential equations (ODE) in which the vector field contains two components, say, one linear and one nonlinear. The linear component typically arises from the diffusion, it might be stiff, and is therefore integrated implicitly. The nonlinear part is easiest to treat with an explicit scheme, and typically comes from the discretization of the convection. The overall method is a time integration scheme called *implicit-explicit* (IMEX). Examples include IMEX Runge-Kutta methods [1, 30, 42], IMEX multistep methods [2, 28], linearly implicit methods (see e.g.[24] and references therein) etc. The time integration methods presented in this thesis are well-related to the IMEX methods in [1, 30], which are a subclass of additive partitioned Runge-Kutta methods.

1.3.4 Commutator-free Lie-group exponential integrators

The time integration method used in this work involves a splitting of the nonlinear convection operator and the linear operator. The convection operator is integrated with the use of explicit *commutator-free Lie-group exponential integrators* (CF) presented in [8], while the diffusion operator is solved implicitly. The framework of the CF methods is such that the exponentials can be treated as exact flows of linearized convecting vector fields. These flows can then be computed accurately using semi-Lagrangian methods. We therefore give a brief description of the CF methods.¹

We consider vector fields F on \mathbb{R}^d , to be smooth functions assigning to each point $y \in \mathbb{R}^d$ a tangent vector at y . The vector field F can be expressed in coordinates as

$$F(y) = \sum_{i=1}^d f_i(y) \left. \frac{\partial}{\partial y_i} \right|_y,$$

¹This section is similar to section 4.5.3 in the Appendix of paper III, chapter 4.

Chapter 1. Introduction

to emphasize the fact that $F(y)$ is a tangent vector [51], where $f_i(y)$ are the components of the tangent vector and $\frac{\partial}{\partial y_i}\Big|_y$ is the canonical basis of the tangent space to \mathbb{R}^d at y .

The flow at time t of F through the point y_0 is denoted by

$$y(t) = \exp(tF)y_0, \quad (1.3.4)$$

where $y(t)$ satisfies the differential equation

$$\dot{y}_i(t) = f_i(y(t)), \quad i = 1, \dots, d, \quad y(t_0) = y_0.$$

The *Lie bracket* (or *commutator*) of two vector fields, F and G is a third vector field obtained by applying first F to G as a differential operator, and then subtracting the result of applying G to F . This leads to

$$[F, G] := \sum_{j=1}^d \sum_{i=1}^d \left(f_i \frac{\partial g_j}{\partial y_i} - g_i \frac{\partial f_j}{\partial y_i} \right) \frac{\partial}{\partial y_j}.$$

For vector fields F and G such that

$$f(y) = Ay, \quad g(y) = By,$$

where A and B are $d \times d$ matrices, the Lie bracket of the vector fields is the vector field $[F, G]$ with components Cy , where $C = BA - AB$ is the matrix commutator of B and A . The composition of the flows of two vector fields F and G can be expressed as the flow of a third vector field defined by means of a series of iterated Lie brackets of the two vector fields F and G . The set of vector fields on \mathbb{R}^d , written $\mathcal{X}(\mathbb{R}^d)$, forms a Lie algebra. A set of vector fields $\{\mathcal{E}_1, \dots, \mathcal{E}_m\}$, $d \leq m$, is a set of *frame vector fields* on \mathbb{R}^d if

$$\mathbb{R}^d = \text{span}\{\mathcal{E}_1|_x, \dots, \mathcal{E}_m|_x\}, \quad \forall x \in \mathbb{R}^d.$$

Given any vector field $F \in \mathcal{X}(\mathbb{R}^d)$ we have

$$F(y) = \sum_{i=1}^m f_i(y) \mathcal{E}_i(y).$$

We denote by F_p the vector field

$$F_p(x) = \sum_{i=1}^m f_i(p) \mathcal{E}_i(x)$$

and we say that F_p is the vector field F *frozen* at the point p .

We define Runge-Kutta commutator-free methods [16, 8] for approximating the flow (1.3.4) as follows:

Algorithm 1.1. *Commutator-free method*

$p = y_n$
for $r = 1 : s$ **do**
 $Y_r = \exp\left(\sum_{k=1}^{r-1} \alpha_{r,l}^k F_k\right) \cdots \exp\left(\sum_{k=1}^{r-1} \alpha_{r,1}^k F_k\right) p$
 $F_r = hF_{Y_r} = h \sum_{i=1}^m f_i(Y_r) \mathcal{E}_i$
end
 $y_{n+1} = \exp\left(\sum_{k=1}^s \beta_J^k F_k\right) \cdots \exp\left(\sum_{k=1}^s \beta_1^k F_k\right) p$

Here n counts the number of time steps and h is the step-size of integration. The integrator has s stages and parameters $\alpha_{r,l}^k, \beta_l^k, r, k = 1, \dots, s$ and $l = 1, \dots, J$. Each new stage value is obtained as a composition of J exponentials (i.e. exact flows) of linear combinations of vector fields frozen at the previously computed stage values. In the case $\mathcal{E}_i(y) = \frac{\partial}{\partial y_i}$ the commutator-free methods reduce to the usual Runge-Kutta methods and the exponentials are simple translations in \mathbb{R}^d . Note that composition of exponentials can be replaced by truncated series of repeated commutators, but this is not always preferable from the point of view of computational complexity.

Now suppose the vector field $F(y)$ represents the semi-discrete nonlinear convection operator in (1.1.1), for a chosen computational grid, and for the sake of simplicity, suppose $v = 0$ and $\mathbf{f} = 0$, then we can apply Algorithm 1.1. The frozen vector fields correspond to semi-discrete, linear, pure convection operators and each of the exponentials in Algorithm 1.1 can be computed in a semi-Lagrangian fashion. More precisely, assume q is an obtained intermediate approximation of the solution known on the computational grid, and so are the stage values $Y_k, k = 1, \dots, r-1$, then each of the exponentials

$$\exp\left(\sum_{k=1}^{r-1} \alpha_{r,l}^k F_k\right) q, \quad (1.3.5)$$

is computed following the procedure outlined in section 1.3.1, by first tracing characteristics solving (1.3.3) backward in time on the interval $[t_n, t_n + h]$, and then interpolating q at the departure points of the characteristics $\tilde{\mathbf{x}} = \chi(t_n)$. The convecting vector field \mathbf{V} of (1.3.3) corresponding to such exponential is the linear combination of the stage values with coefficients $\alpha_{r,l}^k$, and its evaluations in the numerical integration of (1.3.3) require interpolation.

1.3.5 Krylov subspace methods and matrix exponentials

When the vector field $F(y)$ represents a semi-discrete convection operator, the flow (1.3.4) or (1.3.5) could also be straightforwardly computed as the product of a matrix exponential and a vector. Efficient numerical methods for computing such products are the Krylov projection methods [9, 26, 48, 37, 33]. We explore this option in the first paper, section 2.3.1, and in the third paper, section 4.3.3. In general the semi-Lagrangian approach was shown to perform better than the Krylov methods in the solution of convection-dominated convection-diffusion problems. Thus the semi-Lagrangian approach has been the method of choice throughout this thesis. Another option is, for example, the accurate solution of pure convection flows using an order 4 Runge-Kutta integration method. This choice was advocated by Maday *et al.* [34].

1.3.6 Notations

Abbreviations such as RK, IMEX, CF, DIRK, SL etc have meanings as summarized in Table 1.1.

Unless otherwise specified, we denote by h the time step Δt of an integration method. Since the methods presented and analyzed are typically methods for the time integration, the *order* of accuracy when used without specifying would refer to the *temporal order* of convergence. The temporal domain $0 < t < T$ is uniformly discretized into subdomains $t_0 := 0 < t_1 < \dots < t_n < \dots < t_N =: T$, with $t_{n+1} - t_n = h$ for all $n = 0, \dots, N - 1$. We have used **boldface** letters for d -dimensional field variables (where $d = 2$ or 3), thereby distinguishing them from corresponding scalar field variables. However, this rule is not followed for vector fields of arbitrary dimension. All matrices are written with uppercase letters.

Table 1.1: Some important abbreviations

Abbreviation	Meaning
RK	Runge-Kutta
SL	Semi-Lagrangian
IMEX	Implicit-Explicit
DIRK	Diagonally implicit Runge-Kutta
CF	Commutator-free exponential integrators
PDE	Partial differential equations
ODE	Ordinary differential equations
DAE	Differential-algebraic equations

1.4 Summary of results

Paper I: Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

Elena Celledoni and Bawfeh Kingsley Kometa
Published in Journal of Scientific Computing (2009)

Methods for convection-dominated convection-diffusion PDEs are derived. Numerical treatment of the flows of the convection vector fields via semi-Lagrangian methods are explained and the advantage over Eulerian methods are demonstrated numerically. Order conditions for methods of orders 1 to 3 are also presented. Examples for the methods are given. These examples are constructed out of *additive* IMEX partitioned RK methods with DIRK parts. The related coefficients of the CF methods [8] are constructed out of the explicit part of the IMEX RK. The convection term is treated with the CF methods while the linear/diffusion term is treated with the DIRK methods. The overall method is termed DIRK-CF. Using the viscous Burgers' and KdV equations various numerical experiments are carried out to demonstrate

- the advantage of the DIRK-CF over the IMEX methods, and
- the advantage of the SL over Krylov implementations of the DIRK-CF methods (for convection-dominated problems)

over a range of viscosity parameters while keeping the spatial accuracy fixed. The spatial discretization used is the centered-differences for both the convection and diffusion/dispersion operators. Both cubic spline and piecewise cubic Hermite interpolations are employed for the SL methods. The result shows that the SL implementations of the DIRK-CF outperforms the Eulerian methods, at smaller viscosity regimes. It is also observed that the choice of the interpolation method plays a crucial role in the performance of the SL methods.

Paper II: Order conditions for Runge-Kutta exponential integrators for convection-dominated problems

Elena Celledoni and Bawfeh Kingsley Kometa
Preprint series: Numerics, No.4, IMF, NTNU (2009)

In this paper the order conditions for the class of methods presented in Paper I are studied. In paper I it was shown that for methods of order 1 to 3, the order conditions are a subset of order conditions for the CF methods and the parent partitioned RK method from which they are derived. However, we show that for methods of order 4 in this class, there exist extra coupling conditions. Part of the calculations done here had been given in paper I. A new method is then constructed out of the fourth order IMEX method of Kennedy and

Chapter 1. Introduction

Carpenter [30], by enforcing the conditions of order four for the CF methods [41], and the obtained extra coupling conditions. Using the viscous Burgers' equation, we verify numerically that the method achieves order 4.

Paper III: Semi-Lagrangian multistep exponential integrators for index 2 differential-algebraic systems

Elena Celledoni and Bawfeh Kingsley Kometa
Published in Journal of Computational Physics (2011)

IMEX multistep methods based on BDF schemes have been developed and applied for the time discretization of convection-diffusion PDE problems such as the Burgers' equations (see for example [2]) as well as the incompressible Navier-Stokes equations (see [35, 29, 27, 43, 54, 20]).

In this paper we propose a new class of exponential integrators for semi-implicit index 2 DAEs based on the backward differentiation formula (BDF). We name these methods *BDF-CF* for short. They have about the same implementation ease as the DIRK-CF methods introduced in Paper I, and as such they can be regarded as their multistep counterpart. Their main advantage compared to the DIRK-CF is that they can achieve order of convergence higher than 2, when applied to the DAEs both in the algebraic and differential variables.

The methods are a subclass of the IMEX multistep methods and they are closely related to the SBDF methods presented and studied in [2, 28]. Compared to these methods the BDF-CF methods can be used without modifications in a semi-Lagrangian approach to convection-diffusion problems, whereby the exponentials must be realized as flows of pure convection problems.

Paper IV: On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems

Bawfeh Kingsley Kometa
Submitted to Proceedings of the ICNAAM conference, 2011

In this paper we study the relations between CF methods and the semi-Lagrangian discontinuous Galerkin methods (SLDG) presented in [47]. We propose an approach to improve spatial accuracy and to obtain higher temporal-order in the context of SLDG methods. Preliminary numerical tests are presented for the case of linear advection in 1D and 2D. Comparisons are made with the RKDG methods of Cockburn and Shu [12] demonstrating the good performance of the methods at high Courant numbers.

Paper V: Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

Elena Celledoni, Bawfeh Kingsley Kometa and Olivier Verdier
Preprint series: Numerics, No.7, IMF, NTNU (2011)

Direct applications of high order DIRK-CF methods as presented in [7] to the incompressible Navier-Stokes equations were found to yield a loss in order of convergence. The DIRK-CF methods are exponential integrators arising from the IMEX Runge-Kutta techniques proposed in [1], and are semi-Lagrangian when applied to convection diffusion equations. As discussed in [25], inappropriate implementation of projection methods for incompressible flows can lead to a loss in the order of convergence. In this paper we recover the full order of the IMEX methods using projections onto the space of divergence-free vector fields and we discuss the difficulties encountered in using similar techniques for the semi-Lagrangian DIRK-CF methods. We finally assess the performance of the semi-Lagrangian DIRK-CF methods for the Navier-Stokes equations in convection-dominated problems.

References

- [1] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri, *Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations*, Appl. Numer. Math. **25** (1997), 151–167.
- [2] U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton, *Implicit-explicit methods for time-dependent partial differential equations*, SIAM J. Numer. Anal. **32** (1995), no. 3, 797–823.
- [3] L. Bonaventura, *An introduction to semi-Lagrangian methods for geophysical scale flows*, Lecture Notes, MOX - Applied Mathematics Laboratory, Department of Mathematics, Politecnico di Milano, (luca.bonaventura@polimi.it), mox.polimi.it/bonavent/archiv/lezlag000.pdf, Italy.
- [4] A. Bourchtein and L. Bourchtein, *Semi-Lagrangian semi-implicit time-splitting scheme for the shallow water equations*, Internat. J. Numer. Methods Fluids **54** (2007), no. 4, 453–471. MR 2314751 (2008a:76110)
- [5] E. Celledoni, *Eulerian and semi-Lagrangian commutator-free exponential integrators*, CRM Proceedings and Lecture Notes **39** (2005), 77–90.
- [6] E. Celledoni, *Semi-Lagrangian methods and new integrators for convection dominated problems*, Oberwolfach Reports **12** (2006).

Chapter 1. Introduction

- [7] E. Celledoni and B. K. Kometa, *Semi-Lagrangian Runge-Kutta exponential integrators for convection dominated problems*, J. Sci. Comput. **41** (2009), no. 1, 139–164.
- [8] E. Celledoni, A. Marthinsen, and B. Owren, *Commutator-free Lie group methods*, FGCS **19** (2003).
- [9] E. Celledoni and I. Moret, *A Krylov projection method for systems of ODEs*, Appl. Numer. Math. **24** (1997), no. 2-3, 365–378, Volterra centennial (Tempe, AZ, 1996). MR 1464736 (98g:65061)
- [10] A. Chertock and A. Kurganov, *On a practical implementation of particle methods*, Appl. Numer. Math. **56** (2006), no. 10-11, 1418–1431. MR 2245465 (2007k:65014)
- [11] B. Cockburn, G. E. Karniadakis, and C.-W. Shu, *The development of discontinuous Galerkin methods*, Discontinuous Galerkin methods (Newport, RI, 1999), Lect. Notes Comput. Sci. Eng., vol. 11, Springer, Berlin, 2000, pp. 3–50. MR 1842161 (2002e:65002)
- [12] B. Cockburn and C.-W. Shu, *Runge-Kutta discontinuous Galerkin methods for convection-dominated problems*, J. Sci. Comput. **16** (2001), no. 3, 173–261.
- [13] C. J. Cotter, J. Frank, and S. Reich, *Hamiltonian particle-mesh method for two-layer shallow-water equations subject to the rigid-lid approximation*, SIAM J. Applied Dynamical Systems **3** (2004), 69–83.
- [14] C. J. Cotter, J. Frank, and S. Reich, *The remapped particle-mesh semi-Lagrangian advection scheme*, Quarterly Journal of the Royal Meteorological Society **133** (2007), no. 622, 251–260.
- [15] R. Courant, E. Isaacson, and M. Rees, *On the solution of nonlinear hyperbolic differential equations by finite differences*, Comm. Pure. Appl. Math. **5** (1952), 243–255. MR 0053336 (14,756e)
- [16] P. E. Crouch and R. Grossman, *Numerical integration of ordinary differential equations on manifolds*, J. Nonlinear Sci. **3** (1993), no. 1, 1–33. MR 1216986 (94e:65069)
- [17] N. Crouseilles, M. Mehrenberger, and E. Sonnendrücker, *Conservative semi-Lagrangian schemes for Vlasov equations*, J. Comput. Phys. **229** (2010), no. 6, 1927–1953. MR 2586230 (2010j:76092)
- [18] R. E. Ewing and H. Wang, *A summary of numerical methods for time-dependent advection-dominated partial differential equations*, J. Comput. Appl. Math. **128** (2001), no. 1-2, 423–445, Numerical analysis 2000, Vol. VII, Partial differential equations. MR 1820884 (2002a:65003)
- [19] M. Falcone and R. Ferretti, *Convergence analysis for a class of high-order semi-*

-
- Lagrangian advection schemes*, SIAM Journal on Numerical Analysis **35** (1998), no. 3, 909–940.
- [20] P. F. Fischer, G. W. Kruse, and F. Loth, *Spectral element method for transitional flows in complex geometries*, J. Sc. Comput. **17** (2002), no. 1–4, 81–98.
- [21] F. N. Fritsch and R. E. Carlson, *Monotone piecewise cubic interpolation*, SIAM Journal on Numerical Analysis **17** (1980), no. 2, 238–246.
- [22] P.F. Fischer F.X. Giraldo, J.B. Perot, *A spectral element semi-lagrangian (SESL) method for the spherical shallow water equations*, J. Comput. Phys. **190**.
- [23] F. X. Giraldo, *The Lagrange-Galerkin spectral element method on unstructured quadrilateral grids*, J. Comput. Phys. **147** (1998), no. 1, 114–146. MR 1657765 (99j:65177)
- [24] I. Grooms and K. Julien, *Linearly implicit methods for nonlinear PDEs with linear dispersion and dissipation*, J. Comput. Phys. **230** (2011), no. 9, 3630–3650. MR 2780482
- [25] J. L. Guermond, P. Mineev, and Jie Shen, *An overview of projection methods for incompressible flows*, Comput. Methods Appl. Mech. Engrg. **195** (2006), no. 44-47, 6011–6045. MR 2250931 (2007g:76157)
- [26] M. Hochbruck and Ch. Lubich, *On Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal. **34** (1997), no. 5, 1911–1925. MR 1472203 (98h:65018)
- [27] S. Hugues and A. Randriamampianina, *An improved projection scheme applied to pseudospectral methods for the incompressible Navier-Stokes equations*, Internat. J. Numer. Methods Fluids **28** (1998), no. 3, 501–521.
- [28] W. Hundsdorfer and S. J. Ruuth, *IMEX extensions of linear multistep methods with general monotonicity and boundedness properties*, J. Comput. Phys. **225** (2007), no. 2, 2016–2042.
- [29] G. E. Karniadakis, M. Israeli, and S. A. Orszag, *High-order splitting methods for the incompressible Navier-Stokes equations*, J. Comput. Phys. **97** (1991), no. 2, 414–443.
- [30] C. A. Kennedy and M. H. Carpenter, *Additive Runge-Kutta schemes for convection-diffusion-reaction equations*, Appl. Numer. Math. **44** (2003), no. 1-2, 139–181.
- [31] A. T. Layton, *A semi-Lagrangian collocation method for the shallow water equations on the sphere*, SIAM J. Sci. Comput. **24** (2003), no. 4, 1433–1449.
- [32] P. Lin, K. W. Morton, and E. Süli, *Euler characteristic Galerkin scheme with recovery*, RAIRO Modél. Math. Numér. **27** (1993), no. 7, 863–894. MR 1249456 (94m:65162)

Chapter 1. Introduction

- [33] L. Lopez and V. Simoncini, *Analysis of projection methods for rational function approximation to the matrix exponential*, SIAM J. Numer. Anal. **44** (2006), no. 2, 613–635 (electronic). MR 2218962 (2007b:65036)
- [34] Y. Maday, D. Meiron, A. T. Patera, and E. M. Rønquist, *Analysis of iterative methods for the steady and unsteady Stokes problem: Application to spectral element discretizations*, J. Sci. Comput **14** (1993), no. 2, 310–337.
- [35] Y. Maday, A. T. Patera, and E. M. Rønquist, *An operator-integration-factor splitting method for time-dependent problems: application to incompressible fluid flow*, J. Sci. Comput. **5** (1990), no. 4, 263–292.
- [36] B. V. Minchev and W. M. Wright, *A review of exponential integrators for first order semi-linear problems*, Preprint series: Numerics, 02/2005, Department of Mathematics, NTNU, Trondheim, Norway, <http://www.math.ntnu.no/preprint/numerics/2005/N2-2005.pdf>, 2005.
- [37] I. Moret and P. Novati, *RD-rational approximations of the matrix exponential*, BIT **44** (2004), no. 3, 595–615. MR 2106019 (2005h:65073)
- [38] K. W. Morton and A. Priestly, *On characteristic Galerkin and Lagrange Galerkin methods*, Numerical analysis (Dundee, 1985), Pitman Res. Notes Math. Ser., vol. 140, Longman Sci. Tech., Harlow, 1986, pp. 157–172. MR 873108 (87m:65146)
- [39] K. W. Morton, E. Süli, and P. Lin, *Characteristic Galerkin methods for hyperbolic problems*, Nonlinear hyperbolic problems: theoretical, applied, and computational aspects (Taormina, 1992), Notes Numer. Fluid Mech., vol. 43, Vieweg, Braunschweig, 1993, pp. 430–439. MR 1262393
- [40] R. Nair, J. Côté, and A. Stanforth, *Monotonic cascade interpolation for semi-Lagrangian advection*, Quarterly Journal of the Royal Meteorological Society **125** (1999), no. 553, 197–212.
- [41] B. Owren, *Order conditions for commutator-free Lie group methods*, J. Phys. A **39** (2006), no. 19, 5585–5599. MR 2220777 (2007e:37054)
- [42] L. Pareschi and G. Russo, *Implicit-Explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation*, J. Sci. Comput. **25** (2005), no. 1-2, 129–155.
- [43] R. Peyret, *Spectral methods for incompressible viscous flow*, Appl. Math. Sc., vol. 148, Springer, Berlin, 2000.
- [44] O. Pironneau, *On the transport-diffusion algorithm and its applications to the Navier-Stokes equations*, Numer. Math. **38** (1981/82), no. 3, 309–332. MR 654100 (83d:65258)
- [45] O. Pironneau, J. Liou, and T. Tezduyar, *Characteristic-Galerkin and Galerkin/least-squares space-time formulations for the advection-diffusion equations with time-*

-
- dependent domains*, Comput. Methods Appl. Mech. Engrg. **100** (1992), no. 1, 117–141. MR 1187633 (93j:65150)
- [46] P. J. Rasch and D. L. Williamson, *On shape-preserving interpolation and semi-Lagrangian transport*, SIAM J. Sci. Statist. Comput. **11** (1990), no. 4, 656–687. MR 1054632 (91f:65020)
- [47] M. Restelli, L. Bonaventura, and R. Sacco, *A semi-Lagrangian discontinuous Galerkin method for scalar advection by incompressible flows*, J. Comput. Phys. **216** (2006), no. 1, 195–215.
- [48] R. B. Sidje, *Expokit: a software package for computing matrix exponentials*, ACM Trans. Math. Softw. **24** (1998), 130–156.
- [49] P. K. Smorlarkiewicz and G. A. Grell, *A class of monotone interpolation schemes*, J. Comput. Phys. **101** (1992).
- [50] A. Staniforth and J. Côté, *semi-Lagrangian integration schemes for atmospheric models*, Monthly Weather Review **119** (1991), 2206–2223.
- [51] L. W. Tu, *An introduction to manifolds*, second ed., Universitext, Springer, New York, 2011. MR 2723362 (2011g:58001)
- [52] R. A. Walters, E.M. Lane, and R.F. Henry, *semi-Lagrangian methods for a finite element coastal ocean model*, Ocean Modelling **19** (2007), no. 3-4, 112–124.
- [53] A. Wiin-Nielsen, *On the application of trajectory methods in numerical forecasting*, Tellus **11** (1959), no. 2, 180–196.
- [54] D. Xiu and G. E. Karniadakis, *A semi-Lagrangian high-order method for Navier-Stokes equations*, J. Comput. Phys. **172** (2001), no. 2, 658–684. MR 1857617 (2002g:76077)
- [55] M. Zerroukat, N. Wood, and A. Staniforth, *SLICE: A semi-Lagrangian inherently conserving and efficient scheme for transport problems*, Quarterly Journal of the Royal Meteorological Society **128** (2002), no. 586, 2801–2820.

Paper I

Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

Elena Celledoni and Bawfeh Kingsley Kometa

Published in *Journal of Scientific Computing*, Vol. 41 (2009),
pp.139–164

Chapter 2

Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

Abstract. In this paper we consider the case of nonlinear convection-diffusion problems with a dominating convection term and we propose exponential integrators based on the composition of exact pure convection flows. These methods can be applied to the numerical integration of the considered PDEs in a semi-Lagrangian fashion. Semi-Lagrangian methods perform well on convection-dominated problems [Pironneau in *Numer. Math.* 38:309-332; Hockney and Eastwood in *Computer simulations using particles*. McGraw-Hill, New York, 1982; Rees and Morton in *SIAM J. Sci. Stat. Comput.* 12(3):547-572, 1991; Baines in *Moving finite elements*. Monographs on numerical analysis. Clarendon Press, Oxford, 1994].

In these methods linear convective terms can be integrated *exactly* by first computing the characteristics corresponding to the grid points of the adopted discretization, and then producing the numerical approximation via an interpolation procedure.

Key words Additive Runge–Kutta methods, commutator-free methods, convection-diffusion equations, semi-Lagrangian methods

2.1 Introduction

The subject of this work is nonlinear convection-diffusion problems with a dominating convection term:

$$\frac{\partial}{\partial t}u(\mathbf{x}, t) + \mathbf{V}(u)(\mathbf{x}, t) \cdot \nabla u(\mathbf{x}, t) = \nu \nabla^2 u + f(\mathbf{x}), \quad (2.1.1)$$

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

with $\mathbf{x} \in \Omega \subset \mathbb{R}^d$, $\mathbf{V} : \mathbb{R}^d \times [0, T] \rightarrow \mathbb{R}^d$ is a given vector field depending on $u : \mathbb{R}^d \times [0, T] \rightarrow \mathbb{R}$, and $u(\mathbf{x}, 0) = u_0(\mathbf{x})$.

As an example one could consider the Burgers' equation in one space dimension where \mathbf{V} is a scalar function coinciding with u . The unknown u becomes vector valued $\mathbf{u} : \mathbb{R}^d \times [0, T] \rightarrow \mathbb{R}^d$ and $\mathbf{V} = \mathbf{u}$ in the Burgers' equations in two space dimensions.

Another example arises in a popular model for the simulation of internal waves occurring between layers of stratified flow. This model consists of the Navier-Stokes equations coupled with a convection diffusion problem like (2.1.1) modelling transport of temperature or of salt concentration in the fluid. In this case \mathbf{V} is the velocity field solution of the Navier-Stokes equations. This model is called Boussinesq approximation of the Navier-Stokes equations [21]. In order to simulate properly the transport phenomena and the waves it is important to achieve the right balance between convection and diffusion, tuning appropriately ν over repeated numerical experiments.

The case when the parameter ν is very small is particularly interesting and very challenging from the numerical point of view. In the Navier-Stokes equations (which are themselves a suitable generalization of (2.1.1)) this corresponds to the presence of high Reynolds numbers, as for example in the simulation of turbulent flows.

The numerical methods presented in this article are described for the sake of simplicity, but without lack of generality, for the test problem (2.1.1). If we semidiscretize (2.1.1) in space with a finite element or finite difference method we obtain a system of ordinary differential equations of the type

$$y_t - C(y)y = Ay + f, \quad y(0) = y_0. \quad (2.1.2)$$

Here C is the discretized convection operator, A corresponds to the linear diffusion term, often negative definite.

A typical approach for solving numerically the semidiscretized equations is to treat convection and diffusion separately, the diffusion (as it might be stiff) with an implicit approach and the convection with an explicit integrator possibly of higher order, see for example [4, 2, 1, 15]. We will refer to these methods as IMEX methods. This approach has a big advantage, as most of the spatial discretizations of the diffusion operator give rise to matrices which are symmetric and negative definite, and implicit integration of the diffusion would require the solution of only symmetric positive definite linear algebraic systems.

When ν is small enough the diffusion term could in principle be treated explicitly as the CFL condition imposes the use of small time step-sizes anyway¹. In [22] the authors point out that the use of low order semi-implicit methods in the direct numerical simulation of turbulent flows leads to prohibitive time step restrictions. In fact the time-step dictated by the CFL condition can be of several orders of magnitude smaller than the intrinsic temporal scale of the problem predicted by the theory. The CFL condition restrictions can

¹However in this case the iterative solution of the linear systems arising in the IMEX methods would also require just few iterations

be alleviated by the use of semi-Lagrangian schemes. In this case implicit integration for the diffusion term is usually adopted.

We present a new class of integration methods for convection-diffusion problems. These methods are exponential integrators and their peculiarity is that they allow for the computation of exponentials of the linearized convection term of (2.1.2).

A simple example is the following first order integrator for (2.1.2),

$$y_{n+1} = \exp(hC(y_n))y_n + hAy_{n+1} + hf. \quad (2.1.3)$$

The goal here is to present and analyse higher order methods which share some of the features of (2.1.3). The methods in general need not be implicit in the diffusion part, but all the examples given in section 2.2 have this feature. The main reason for developing this type of methods is that as it turns out they can be applied to the numerical integration of the considered PDEs in a semi-Lagrangian fashion. In semi-Lagrangian methods [19, 14, 20, 3], linear convective terms are integrated *exactly* by first computing the characteristics corresponding to the gridpoints of the adopted discretization, and then producing the numerical approximation via an interpolation procedure.

The exponential $\exp(\gamma hC(w)) \cdot g$ (where w and g are fixed vectors in \mathbf{R}^N , and γ is a parameter of the integration method) is the solution of the semidiscretized equation

$$v' = C(w)v, \quad v(0) = g, \quad \text{in } [0, \gamma h], \quad (2.1.4)$$

which corresponds to the pure convection problem

$$\begin{aligned} \varphi_t + \tilde{\mathbf{V}} \cdot \nabla \varphi &= 0, & \varphi(x_i, 0) &= g_i, & \text{in } [0, \gamma h] \times \Omega, & \text{i.e.} \\ \frac{D\varphi}{Dt} &= 0, & \varphi(x_i, 0) &= g_i, & \text{in } [0, \gamma h] \times \Omega, \end{aligned} \quad (2.1.5)$$

where x_i are the nodes of the space discretization, $\tilde{\mathbf{V}}$ is the convecting vector field obtained interpolating the values $\tilde{\mathbf{V}}(x_i) = w_i$, and $\frac{D}{Dt}$ is the total derivative. If we apply (2.1.3) directly to the problem (2.1.1) before discretizing in space, and we make the exponential $\exp(hC(y_n))y_n$ correspond to the exact integration of a pure convection problem, we obtain the following well known transport-diffusion algorithm studied in [19],

$$\begin{aligned} \frac{D\tilde{u}_{n+\frac{1}{2}}}{Dt} &= 0, & \tilde{u}_{n+\frac{1}{2}}(x, t_n) &= u_n(x), & \text{on } [t_n, t_n + h] \\ u_{n+\frac{1}{2}}(x) &:= \tilde{u}_{n+\frac{1}{2}}(x, t_n + h) \\ u_{n+1} &= u_{n+\frac{1}{2}} + h\nu\nabla^2 u_{n+1} + hf, \end{aligned} \quad (2.1.6)$$

the convecting vector field is $\tilde{\mathbf{V}}(x) = u_n(x)$. The accurate integration of the pure convection problem in (2.1.6) can be obtained by introducing characteristics, we get

$$\begin{aligned} u_{n+\frac{1}{2}}(x) &= \tilde{u}_{n+\frac{1}{2}}(x, t_n + h) = u_n(X(t_n)) \\ \frac{dX}{d\tau} &= u_n(X(\tau)), & X(t_n + h) &= x, \end{aligned} \quad (2.1.7)$$

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

and the equation for the characteristics $X(\tau)$ must be integrated backwards in time, either exactly or with a suitable numerical integrator. Now substituting the result in the second equation of (2.1.6) we have

$$\begin{aligned}\frac{dX}{d\tau} &= u_n(X(\tau)), \quad X(t_n + h) = x, \\ u_{n+1} &= u_n(X(t_n)) + h\nu\nabla^2 u_{n+1} + hf.\end{aligned}\tag{2.1.8}$$

The outlined correspondence between (2.1.3) and (2.1.6) motivates the use of exponentials of C (the semidiscrete convection operator of (2.1.2)) in the integrators.

The presented methods are of additive type. We derive methods up to and including order three. Additive Runge-Kutta methods can be cast into the class of partitioned Runge-Kutta methods and their order conditions can be studied in this setting [13, 15]. In particular applying two Runge-Kutta methods of order p to two different parts of the vector field of the problem, does not guarantee order p overall: extra coupling conditions must be satisfied by the coefficients of the two methods. In our methods the convection part is treated by a commutator-free explicit method requiring the composition of exponentials of C [8], the linear diffusion with a Runge-Kutta (implicit) method. For low order one can show that the order conditions are a subset of the union of classical order conditions for partitioned Runge-Kutta methods and commutator-free methods.² As a consequence it is quite easy to derive examples of methods in the new class by taking known IMEX methods as a starting point. In general the order conditions for the new methods do not reduce to a special case of the known order conditions for partitioned Runge-Kutta methods. New coupling conditions, involving the coefficients of the commutator free methods, do appear at order four [7].

One nice feature of these methods compared to analogous IMEX methods is that they inherit the A-stability features of the implicit part of the integrator.

The semi-Lagrangian methods here presented are based on Runge-Kutta one-step strategies and are for this reason an alternative to the semi-Lagrangian multistep approaches proposed for example in [22]. Previous preliminary work has also been presented in [6].

In section 2.2 we present the new methods, discuss stability and the order conditions and give some concrete examples. In section 2.3 we illustrate the performance of the methods implementing them as semi-Lagrangian schemes.

2.2 Presentation of the new class of methods

We consider the semidiscretized ordinary differential equation

$$\dot{y} - C(y)y = Ay, \quad y(0) = y_0,\tag{2.2.1}$$

²See [17] for the order theory for commutator free methods.

where for the sake of simplicity, but without loss of generality, we have dropped the forcing term f of (2.1.2). Here A is a $N \times N$ matrix with constant coefficients and $C(y)$ is a matrix-valued function, also $N \times N$.

The methods here studied have a Runge-Kutta like format with two sets of parameters: $\mathcal{A} = \{a_{i,j}\}_{i,j=1,\dots,s}$, $\mathbf{b} = [b_1, \dots, b_s]$, $\mathbf{c} = [c_1, \dots, c_s]$ and $\alpha_{i,l}^j, \beta_l^j$, $i = 1, \dots, s$, $j = 1, \dots, s$, $l = 1, \dots, J$, $\hat{\mathbf{c}} = [\hat{c}_1, \dots, \hat{c}_s]$. The methods have the following format:

for $i = 1 : s$ do

$$Y_i = \varphi_i y_n + h \sum_{j=1}^s a_{i,j} \varphi_{i,j} A Y_j,$$

$$\varphi_i = \exp(h \sum_k \alpha_{i,J}^k C(Y_k)) \cdots \exp(h \sum_k \alpha_{i,1}^k C(Y_k)),$$

$$\varphi_{i,j} := \varphi_i \varphi_j^{-1}$$

end

$$y_{n+1} = \varphi_{s+1} y_n + h \sum_{i=1}^s b_i \varphi_{n+1,i} A Y_i,$$

$$\varphi_{s+1} = \exp(h \sum_k \beta_s^k C(Y_k)) \cdots \exp(h \sum_k \beta_1^k C(Y_k)),$$

$$\varphi_{s+1,i} := \varphi_{s+1} \varphi_i^{-1}.$$

These methods are associated to the two Butcher tableaus,

$$\begin{array}{c|c} \mathbf{c} & \mathcal{A} \\ \hline & \mathbf{b} \end{array}, \quad \begin{array}{c|c} \hat{\mathbf{c}} & \hat{\mathcal{A}} \\ \hline & \hat{\mathbf{b}} \end{array}, \quad (2.2.2)$$

where we have defined

$$\hat{a}_{i,j} := \sum_{l=1}^J \alpha_{i,l}^j, \quad \hat{b}_j := \sum_{l=1}^J \beta_l^j, \quad (2.2.3)$$

for $i = 1, \dots, s$, and $\hat{\mathcal{A}} = \{\hat{a}_{i,j}\}_{i,j=1,\dots,s}$ and $\hat{\mathbf{b}} = [\hat{b}_1, \dots, \hat{b}_s]$. The coefficients of the first tableau are used for the linear vector field Ay while the coefficients of the second tableau, split up in the sums (2.2.3), are used for the nonlinear vector field $C(y)y$.

For reasons of computational ease in the implementation of the methods, we are particularly interested in the case where the coefficients $a_{i,j}$ define a diagonally implicit Runge-Kutta method (DIRK), so that $a_{i,j} = 0$ when $j > i$. In this case the methods require the solution of only one linear system per stage. The discussion on order conditions is not affected by this choice.

The presented methods can be directly interpreted as transport-diffusion algorithms for convection-diffusion equations.

Remark 2.2.1. Let us consider the following change of variables in (2.2.1)

$$y(t) = W(t) \cdot z(t), \quad (2.2.4)$$

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

with

$$W_t = C(Wz)W, \quad W(0) = I, \quad (2.2.5)$$

I is the identity matrix. After differentiation and substituting for y_t in the right hand side of (2.2.1), we obtain the following system of equations equivalent to (2.2.1)

$$\begin{cases} z_t &= W^{-1}AWz, \quad z(0) = y_0, \\ W_t &= C(Wz)W, \quad W(0) = I. \end{cases} \quad (2.2.6)$$

We apply a partitioned Runge-Kutta approach to this system combining an implicit Runge-Kutta method for the equation for z and an explicit commutator-free method (in short CF) for the equation for W , see Appendix 2.4. We obtain

for $i = 1 : s$ do

$$Z_i = z_n + h \sum_{j=1}^i a_{i,j} Q_j^{-1} A Q_j Z_j$$

$$Q_i = \exp(\sum_k \alpha_{i,j}^k C(Q_k Z_k)) \cdots \exp(\sum_k \alpha_{i,1}^k C(Q_k Z_k)) \cdot W_n$$

end

$$z_{n+1} = z_n + \sum_{i=1}^s b_i Q_i^{-1} A Q_i Z_i$$

$$W_{n+1} = \exp(\sum_k \beta_j^k C(Q_k Z_k)) \cdots \exp(\sum_k \beta_1^k C(Q_k Z_k)) \cdot W_n.$$

Using the following transformation

$$Y_i := Q_i Z_i, \quad \varphi_i := Q_i W_n^{-1}, \quad y_{n+1} = W_{n+1} z_{n+1}, \quad \varphi_{s+1} = W_{n+1} W_n^{-1}, \quad (2.2.7)$$

we obtain the original format of the methods. Since $Q_i Q_j^{-1} = \varphi_i \varphi_j^{-1}$ then $\varphi_{i,j} := \varphi_i \varphi_j^{-1}$ arises naturally in the expression for Y_i .

2.2.1 Stability function

For the stability analysis we follow the approach in [1], namely we apply the method to the linear test equation

$$\dot{y} = \lambda y + \hat{i} \mu y, \quad (2.2.8)$$

where $\lambda < 0$ and μ are real numbers, and \hat{i} is the imaginary unit ($\hat{i}^2 = -1$). Further define $z := v + \hat{i}w$ where $v = \lambda h$ and $w = \mu h$.

We get

$$Y_i = e^{\hat{i}w\hat{c}_i} y_0 + v \sum_{j=1}^s a_{i,j} e^{\hat{i}w(\hat{c}_i - \hat{c}_j)} Y_j, \quad i = 1, \dots, s, \quad (2.2.9)$$

$$y_1 = e^{\hat{i}w} [y_0 + v \sum_{i=1}^s b_i e^{-\hat{i}w\hat{c}_i} Y_i]. \quad (2.2.10)$$

Writing

$$\mathbb{Y} = [Y_1, \dots, Y_s]^T$$

equations (2.2.9) get the form

$$\mathbb{Y} = (I_s - v\mathcal{A}(w))^{-1}D(w)\mathbf{1}_s y_0,$$

where $\mathcal{A}(w)_{ij} = a_{ij}e^{\hat{w}(\hat{c}_i - \hat{c}_j)}$, $D(w)_{ij} = e^{\hat{w}\hat{c}_i}\delta_{ij}$, and $\mathbf{1}_s = [1, \dots, 1]^T$.

From (2.2.10) we get

$$y_1 = e^{\hat{w}}[1 + v\mathbf{b}^T D(w)^{-1}(I_s - v\mathcal{A}(w))^{-1}D(w)\mathbf{1}_s]y_0.$$

Thus the stability function can be expressed as

$$R(v, w) := e^{\hat{w}}[1 + v\mathbf{b}^T D(w)^{-1}(I_s - v\mathcal{A}(w))^{-1}D(w)\mathbf{1}_s]. \quad (2.2.11)$$

But by direct calculation one can see that

$$D(w)^{-1}(I_s - v\mathcal{A}(w))^{-1}D(w) = (I_s - v\mathcal{A})^{-1}.$$

Thus

$$R(v, w) = e^{\hat{w}}\tilde{R}(v),$$

where \tilde{R} is the stability function of the implicit RK method \mathcal{A} , see [13] on stability functions of Runge-Kutta methods. This means the linear stability of our method is determined to a great extent by the stability of the RK method with tableau $\begin{array}{c|c} \mathbf{c} & \mathcal{A} \\ \hline & \mathbf{b} \end{array}$.

This is not the case for the IMEX methods, where the stability function is

$$R(v, w) := 1 + (v\mathbf{b}^T + \hat{w}\hat{\mathbf{b}}^T)(I_s - v\mathcal{A} - \hat{w}\hat{\mathcal{A}})^{-1}\mathbf{1}_s,$$

therefore depending also on the explicit method with tableau $\begin{array}{c|c} \hat{\mathbf{c}} & \hat{\mathcal{A}} \\ \hline & \hat{\mathbf{b}} \end{array}$ and may not be A-stable.

2.2.2 Order conditions

In the study of the order conditions it is convenient to treat the numerical solution y_{n+1} as an extra stage value

$$Y_{s+1} := \varphi_{s+1}y_n + h \sum_{j=1}^s a_{s+1,j}\varphi_{s+1,j}AY_j, \quad a_{s+1,j} := b_j,$$

with

$$\varphi_{s+1} = \exp\left(h \sum_k \alpha_{s+1,k}^k C(Y_k)\right) \cdots \exp\left(h \sum_k \alpha_{s+1,1}^k C(Y_k)\right), \quad \alpha_{s+1,J-l}^k := \beta_{J-l}^k,$$

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

and with $\varphi_{s+1,j} = \varphi_{s+1}\varphi_j^{-1}$.

Assuming enough regularity of the solution of (2.2.1), expanding the exact and the numerical solution with Taylor series and matching the terms one obtains the order conditions reported in Tables 2.4, 2.5.

In the tables we report the elementary differentials appearing in the Taylor expansions, and the corresponding bicoloured binary trees, related to each order condition. Black nodes (\bullet) refer to the vector field $C(y)y$ and white nodes (\circ) to Ay .

Derivatives of the exact solution

The derivatives of the exact solution can be written in the form

$$y^{(q)} = \sum_{k=0}^{q-1} \binom{q-1}{k} \left(\frac{d^{q-1-k}}{dh^{q-1-k}} (C(y) + A) \right) y^{(k)},$$

and therefore

$$\begin{aligned} y^{(1)} \Big|_{h=0} &= \dot{y}|_{h=0} = C(y_0)y_0 + Ay_0, \\ y^{(2)} \Big|_{h=0} &= C'(y_0)(\dot{y}(0))y_0 + (C(y_0) + A)^2y_0. \end{aligned}$$

Here and in the sequel the upper index within round brackets denotes the order of differentiation. We have used the matrix $C'(y)(w)$ obtained by differentiating $C(y)$,

$$(C'(y)(w))_{i,j} := \sum_{k=1}^N \frac{\partial c_{i,j}}{\partial x_k}(y)w_k,$$

and $c_{i,j} := (C(y))_{i,j}$, and N is the number of components of y .

In short we will write $C, C'(\cdot)$ for $C(y_0), C'(y_0)(\cdot)$ respectively.

Derivatives of the numerical solution

We assume without loss of generality that $n = 0$.

We obtain

$$\dot{Y}_{s+1} \Big|_{h=0} = \dot{\varphi}_{s+1} \Big|_{h=0} y_0 + \sum_{j=1}^s a_{s+1,j} A y_0$$

In general we have

$$\begin{aligned} Y_i^{(q)} \Big|_{h=0} &= \varphi_i^{(q)} \Big|_{h=0} y_0 + q \frac{d^{q-1}}{dh^{q-1}} \left(\sum_j a_{i,j} \varphi_i \varphi_j^{-1} A Y_j \right) \Big|_{h=0} \\ &= \varphi_i^{(q)} \Big|_{h=0} y_0 + q \sum_j a_{i,j} \sum_{k=0}^{q-1} \binom{q-1}{k} \varphi_{i,j}^{(q-1-k)} A Y_j^{(k)} \Big|_{h=0}, \end{aligned} \quad (2.2.12)$$

where $\varphi_{i,j} := \varphi_i \varphi_j^{-1}$, $i, j = 1, \dots, s$.

The equations

$$Y_{s+1}^{(q)} \Big|_{h=0} = y^{(q)} \Big|_{h=0}, \quad q = 0, 1, \dots, p$$

give the order conditions for order p , and for $p = 1, 2, 3$ are reported in Table 2.4 (page 31) and Table 2.5 (page 32).

The recursive computation of the derivatives of Y_i requires the use of φ_i and $\varphi_{i,j}$ and their derivatives which we will now discuss.

Consider the following matrix-valued functions,

$$C_{i,J-l} := h \sum_k \alpha_{i,J-l}^k C(Y_k), \quad l = 0, \dots, J-1,$$

and

$$\tilde{C}_{i,J-l} := -h \sum_k \alpha_{i,l+1}^k C(Y_k), \quad l = 0, \dots, J-1.$$

Consider also the following product of matrix exponentials

$$\psi_i := \exp(B_{i,J}) \cdot \exp(B_{i,J-1}) \cdots \exp(B_{i,1}),$$

such that

$$\psi_i = \begin{cases} \varphi_i, & B_{i,J-l} = C_{i,J-l}, \quad l = 0, \dots, J-1 \\ \text{and} \\ \varphi_i^{-1} & B_{i,J-l} = \tilde{C}_{i,J-l}, \quad l = 0, \dots, J-1, \end{cases}$$

respectively. We will make use of

$$\psi_i^l := \exp(B_{i,J}) \cdot \exp(B_{i,J-1}) \cdots \exp(B_{i,J-l}).$$

We obtain¹

$$\dot{\psi}_i = \sum_{l=0}^{J-1} \text{Ad}_{\psi_i^l} \left(\text{dexp}_{-B_{i,J-l}}(\dot{B}_{i,J-l}) \right) \cdot \psi_i,$$

so

$$\dot{\psi}_i = S_i(h) \psi_i, \quad S_i(h) := \sum_{l=0}^{J-1} \text{Ad}_{\psi_i^l} \left(\text{dexp}_{-B_{i,J-l}}(\dot{B}_{i,J-l}) \right),$$

and as a direct consequence we have

$$\psi_i^{(r)} = \sum_{k=0}^{r-1} \binom{r-1}{k} \left(\frac{d^{r-1-k}}{dh^{r-1-k}} S_i(h) \right) \psi_i^{(k)}. \quad (2.2.13)$$

The following proposition will be used to find the derivatives of $S_i(h)$.

¹We recall that $\text{dexp}_w(u) := \left. \frac{e^z - 1}{z} \right|_{z=\text{ad}_w} (u) = u + 1/2![w, u] + 1/3![w, [w, u]] + \dots$ and $\text{ad}_w(u) := [w, u]$ (matrix commutator between w and u), $\text{Ad}_\psi(u) := \psi u \psi^{-1}$.

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

Proposition 2.2.1. Given $Z^0 = Z^0(h)$, $W = W(h)$ two matrix-valued differentiable functions then

$$\frac{d^r}{dh^r} \text{Ad}_W Z^0 = \text{Ad}_W Z^r, \quad (2.2.14)$$

with

$$Z^r = [W^{-1} \dot{W}, Z^{r-1}] + \dot{Z}^{r-1}. \quad (2.2.15)$$

Proof. By induction. □

From (2.2.15) by differentiation we obtain

$$\dot{Z}^r = [W^{-1} \dot{W} + W^{-1} \ddot{W}, Z^{r-1}] + [W^{-1} \dot{W}, \dot{Z}^{r-1}] + \ddot{Z}^{r-1}, \quad (2.2.16)$$

and using (2.2.15) and (2.2.16) and assuming $W(0) = I$, we obtain

$$\begin{aligned} \left. \frac{d}{dh} \text{Ad}_W Z^0 \right|_{h=0} &= Z^1(0) = [\dot{W}(0), Z^0(0)] + \dot{Z}^0(0) \\ &\quad \dot{Z}^1(0) = [-\dot{W}(0)^2 + \ddot{W}(0), Z^0(0)] + [\dot{W}(0), \dot{Z}^0(0)] + \ddot{Z}^0(0) \\ \left. \frac{d^2}{dh^2} \text{Ad}_W Z^0 \right|_{h=0} &= Z^2(0) = [\dot{W}(0), Z^1(0)] + \dot{Z}^1(0) \\ &\quad \dot{Z}^2(0) = [-\dot{W}(0)^2 + \ddot{W}(0), Z^1(0)] + [\dot{W}(0), \dot{Z}^1(0)] + \ddot{Z}^1(0) \\ \left. \frac{d^3}{dh^3} \text{Ad}_W Z^0 \right|_{h=0} &= Z^3(0) = [\dot{W}(0), Z^2(0)] + \dot{Z}^2(0) \\ &\quad \vdots \end{aligned} \quad (2.2.17)$$

Further assuming $Z^0 = \text{dexp}_{-B}(\dot{B})$ for some matrix-valued differentiable function $B = B(h)$, expanding the right hand side and differentiating we obtain

$$\begin{aligned} Z^0(0) &= \dot{B}(0) \\ \dot{Z}^0(0) &= \ddot{B}(0) \\ \dot{Z}^1(0) &= \ddot{B}(0) - \frac{1}{2}[\dot{B}(0), \ddot{B}(0)] \\ \dot{Z}^2(0) &= B^{(4)}(0) - [\dot{B}(0), \ddot{B}(0)] + \frac{1}{2}[\dot{B}(0), [\dot{B}(0), \ddot{B}(0)]] \\ &\quad \vdots \end{aligned} \quad (2.2.18)$$

We can now obtain the derivatives of S_i and ψ_i . By setting $W = \psi_i^l$ and $B = B_{i,J-l}$ we can calculate the derivatives of S_i using (2.2.14). We obtain

$$\begin{aligned} S_i(0) &= \sum_{l=0}^{J-1} \dot{B}_{i,J-l}, \\ \left. \frac{d}{dh} S_i \right|_{h=0} &= \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} [\dot{B}_{i,J-r}(0), \dot{B}_{i,J-l}(0)] + \ddot{B}_{i,J-l}(0), \\ \left. \frac{d^2}{dh^2} S_i \right|_{h=0} &= 2 \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} [\dot{B}_{i,J-r}(0), \ddot{B}_{i,J-l}(0)] + \\ &\quad \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} [\ddot{B}_{i,J-r}(0), \dot{B}_{i,J-l}(0)] + \\ &\quad \sum_{l=0}^{J-1} (\ddot{B}_{i,J-l}(0) - \frac{1}{2}[\dot{B}_{i,J-l}(0), \dot{B}_{i,J-l}(0)]), \end{aligned} \quad (2.2.19)$$

analogously the derivatives of $S_i^l = \sum_{r=0}^{J-l-1} \text{Ad}_{\psi_i^r}(\text{dexp}_{-B_{i,J-r}} \dot{B}_{i,J-r})$ are obtained as in the previous formulae, but substituting $J-1$ as upper index in the external summation with $J-l-1$.

In Table 2.2 we report the values of the derivatives of φ_i and φ_j^{-1} at $h=0$, these are obtained from (2.2.13) using (2.2.19). In Table 2.3 we report the derivatives at 0 of $\varphi_{i,j}$, which are obtained using Table 2.2 and

$$\varphi_{i,j}^{(m)} = \sum_{r=0}^m \binom{m}{r} \varphi_i^{(m-r)} (\varphi_j^{-1})^{(r)}.$$

The derivatives of Y_i are reported in Table 2.1 and are obtained using Tables 2.2 and 2.3 and the recursion formula (2.2.12).

We here derive the first few rows of Table 2.2, Table 2.3 and Table 2.1. From (2.2.13) we obtain that

$$\dot{\varphi}_i(0) = \sum_{l=0}^{J-1} \dot{B}_{i,J-l}(0) = \sum_{l=0}^{J-1} \dot{C}_{i,J-l}(0) = \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k C = \left(\sum_k \hat{a}_{i,k} \right) C, \quad (2.2.20)$$

analogously one can compute $\dot{\varphi}_j^{-1}(0)$. These expressions are reported in Table 2.2 and can be used in (2.2.12) to obtain

$$\dot{Y}_i|_{h=0} = \left(\sum_{k=1}^s \hat{a}_{i,k} \right) C y_0 + \left(\sum_{j=1}^s a_{i,j} \right) A y_0 = \hat{c}_i C y_0 + c_i A y_0, \quad (2.2.21)$$

see Table 2.1. Imposing $\dot{Y}_{s+1}|_{h=0} = \dot{y}(0)$, where $\dot{y}(0) = C y_0 + A y_0$ we obtain the following order conditions for order 1,

$$\sum_{k=1}^s \hat{a}_{s+1,k} = 1, \quad \sum_{j=1}^s a_{s+1,j} = 1.$$

These correspond to requiring consistency of the two Runge-Kutta methods (2.2.2).

For deriving the conditions for order two we use (2.2.12) and we get

$$\ddot{Y}_i|_{h=0} = \varphi_i^{(2)}|_{h=0} y_0 + 2 \sum_j a_{i,j} \left(\varphi_{i,j}^{(1)}(0) A Y_j(0) + \varphi_{i,j}(0) A \dot{Y}_j(0) \right)$$

with $\varphi_{i,j}(0) = I$ and $\varphi_{i,j}^{(1)}(0) = \dot{\varphi}_i(0) - \dot{\varphi}_j(0) = (\hat{c}_i - \hat{c}_j)C$. Using $\dot{Y}_j(0)$ from Table 2.1 we obtain

$$\ddot{Y}_i|_{h=0} = \varphi_i^{(2)}|_{h=0} y_0 + 2 \sum_j a_{i,j} ((\hat{c}_i - \hat{c}_j)C A y_0 + \hat{c}_j A C y_0 + c_j A^2 y_0). \quad (2.2.22)$$

From (2.2.13) and (2.2.19) we obtain

$$\begin{aligned} \varphi_i^{(2)}|_{h=0} &= \frac{dS_i(h)}{dh} \varphi_i(h)|_{h=0} + (S_i(h)^2 \varphi_i(h))|_{h=0} \\ &= 2 \sum_k \hat{a}_{i,k} \hat{c}_k C' (C y_0) + 2 \sum_k \hat{a}_{i,k} c_k C' (A y_0) + \left(\sum_{j=1}^s \hat{a}_{i,j} \right)^2 C^2, \end{aligned} \quad (2.2.23)$$

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

Table 2.1: Derivatives of Y_i at 0.

q	$Y_i^{(q)}(0)$
0	y_0
1	$(\sum_j \hat{a}_{i,j})Cy_0 + (\sum_j a_{i,j})Ay_0$
2	$2 \sum_j \hat{a}_{i,j}C'(\hat{c}_jC + c_jA) + \hat{c}_i^2C^2 + 2\hat{c}_i c_i CA - 2(\sum_j a_{i,j}\hat{c}_j)CA + 2 \sum_j a_{i,j}A(\hat{c}_jC + c_jA)$

Table 2.2: Derivatives of φ_i and its inverse.

q	$\varphi_i^{(q)}(0)$	$(\varphi_j^{-1})^{(q)}(0)$
0	I	I
1	$C\hat{c}_i$	$-C\hat{c}_j$
2	$2 \sum_k \hat{a}_{i,k}C'(\hat{c}_kCy_0 + c_kAy_0) + \hat{c}_i^2C^2$	$-2 \sum_k \hat{a}_{j,k}C'(\hat{c}_kCy_0 + c_kAy_0) + \hat{c}_j^2C^2$

Table 2.3: Derivatives of $\varphi_{i,j}$.

q	$\varphi_{i,j}^{(q)}(0)$
0	I
1	$C(\hat{c}_i - \hat{c}_j)$
2	$2 \sum_k (\hat{a}_{i,k} - \hat{a}_{j,k})C'(\hat{c}_kCy_0 + c_kAy_0) + (\hat{c}_i - \hat{c}_j)^2C^2$

which is reported in Table 2.2. Substituting the results in (2.2.22) we obtain

$$\begin{aligned} \ddot{Y}_i|_{h=0} &= (2 \sum_k \hat{a}_{i,k}\hat{c}_kC'(Cy_0) + 2 \sum_k \hat{a}_{i,k}c_kC'(Ay_0) + \hat{c}_i^2C^2)y_0 + \\ & 2 \sum_j a_{i,j}((\hat{c}_i - \hat{c}_j)CAy_0 + \hat{c}_jACy_0 + c_jA^2y_0), \end{aligned} \quad (2.2.24)$$







and report it in Table 2.1.

Taking $i = s + 1$ and matching this result with $y^{(2)}|_{h=0} = C'(\dot{y}(0))y_0 + (C + A)^2y_0$, we obtain the four conditions for order two

$$\begin{aligned} \sum_j a_{s+1,j}\hat{c}_j &= \frac{1}{2}, & \sum_j \hat{a}_{s+1,j}c_j &= \frac{1}{2}, \\ \sum_j a_{s+1,j}c_j &= \frac{1}{2}, & \sum_j \hat{a}_{s+1,j}\hat{c}_j &= \frac{1}{2}. \end{aligned}$$

The conditions for order 3 are obtained in a similar way, see [7].

Table 2.4: Conditions of order 1 and 2.

condition	trees	elementary differential
order 1		
$\sum_i \hat{b}_i = 1$		C
$\sum_i b_i = 1$		A
order 2		
$2 \sum_i \hat{b}_i \hat{c}_i = 1$		$C'(C)$
$2 \sum_i \hat{b}_i c_i = 1$		$C'(A)$
$2 \sum_i b_i \hat{c}_i = 1$		AC
$2 \sum_i b_i c_i = 1$		A^2

It is easy to check, see for example [13], that the conditions for order two coincide with the conditions of partitioned Runge-Kutta methods of the same order for the tableaux (2.2.2). Under the assumption that $c_k = \hat{c}_k$, the conditions for order three (see also [7]) include a subset of the classical conditions of order 3 for partitioned RK methods for (2.2.2). Due to the linearity of A one of the classical elementary differentials does not appear. If we assume $c_k = \hat{c}_k$ the remaining conditions of order three (the first two in Table 2.5) coincide with the conditions for commutator-free methods of order three [8] and [17].

In the tables we have associated to each order condition its corresponding elementary differential and its corresponding bicolored rooted tree. We have used trees τ with $|\tau| + 1$ nodes for representing the differentials of order $|\tau|$ in order to include the commutators between lower order differentials, as done in [17]³.

Remark 2.2.2. We consider the system (2.2.6). Xiu et al. in [22] considered splittings of the Navier-Stokes equations into a convection and a diffusion problem and studied a semi-Lagrangian version of the partitioned combination: the second order Adams-Basforth and the Crank-Nicolson scheme (midpoint rule) (AB2-CN). This method does not achieve second order as a time integrator. Instead they proposed and proved numerically that combining a semi-Lagrangian scheme for the nonlinear convection term (linearized via a second order extrapolation scheme) together with the implicit midpoint rule for the remaining terms, would result in a scheme with overall second order accuracy in time. We have verified their observations in our framework by applying a CF version of the second order Adams-Bashforth to the first equation of (2.2.6) (convection part) and the

³Recall the notation: if τ is a rooted tree $|\tau|$ is the number of nodes of the tree.

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

Table 2.5: Conditions of order 3.

condition	trees	elementary differential
$4 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \beta_{J-r}^k \sum_m \beta_{J-l}^m \hat{c}_m +$ $-2 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \beta_{J-r}^k \sum_m \beta_{J-l}^m \hat{c}_k +$ $- \sum_{l=0}^{J-1} \sum_k \beta_{J-l}^k \sum_r \beta_{J-l}^r \hat{c}_r = 0$		$[C'(C), C]$
$4 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \beta_{J-r}^k \sum_m \beta_{J-l}^m c_m +$ $-2 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \beta_{J-r}^k \sum_m \beta_{J-l}^m c_k +$ $- \sum_{l=0}^{J-1} \sum_k \beta_{J-l}^k \sum_r \beta_{J-l}^r c_r = 0$		$[C'(A), C]$
$3 \sum_k \hat{b}_k \hat{c}_k^2 = 1$		$C''(C, C)$
$3 \sum_k \hat{b}_k \hat{c}_k c_k = 1$		$C''(C, A)$
$3 \sum_k \hat{b}_k c_k^2 = 1$		$C''(A, A)$
$6 \sum_k \hat{b}_k \sum_j \hat{a}_{k,j} \hat{c}_j = 1$		$C'(C'(C))$
$6 \sum_k \hat{b}_k \sum_j \hat{a}_{k,j} c_j = 1$		$C'(C'(A))$
$6 \sum_k \hat{b}_k \sum_j a_{k,j} \hat{c}_j = 1$		$C'(AC)$
$6 \sum_k \hat{b}_k \sum_j a_{k,j} c_j = 1$		$C'(A^2)$
$6 \sum_j b_j c_j - 6 \sum_j b_j \hat{c}_j c_j = 1$		CA^2
$6 \sum_j b_j \sum_m \hat{a}_{j,m} \hat{c}_m = 1$		$AC'(C)$
$6 \sum_j b_j \sum_m \hat{a}_{j,m} c_m = 1$		$AC'(A)$
$3 \sum_j b_j \hat{c}_j^2 = 1$		AC^2
$6 \sum_j b_j \sum_m a_{j,m} \hat{c}_m = 1$		A^2C
$6 \sum_j b_j \sum_m a_{j,m} c_m = 1$		A^3

implicit trapezoidal rule to the second equation of (2.2.6) (diffusion part) of the nonlinear convection-diffusion problem.

Assuming that W_{n-1}, W_n are known, from the first equation of (2.2.6) we get

$$W_{n+1} = \exp\left(\frac{h}{2}(3C(y_n) - C(y_{n-1}))\right)W_n \quad (2.2.25)$$

which agrees with the semi-Lagrangian method used in [22], but in place of the midpoint rule, we apply the implicit trapezoidal rule for the diffusion term to obtain

$$z_{n+1} = z_n + \frac{h}{2}(W_n^{-1}AW_n z_n + W_{n+1}^{-1}AW_{n+1} z_{n+1}). \quad (2.2.26)$$

Transforming back to the variable $y = Wz$ and assuming y_{n-1} and y_n are known apriori we have the following scheme

$$y_{n+1} = E_n y_n + \frac{h}{2}(E_n A y_n + A y_{n+1}), \text{ where } E_n = \exp\left(\frac{h}{2}(3C(y_n) - C(y_{n-1}))\right) \quad (2.2.27)$$

which is tested numerically to be second order accurate in time. This has also been verified directly by Taylor expansion. A direct translation of the AB2-CN scheme into our semi-Lagrangian method results in

$$y_{n+1} = E_n y_n + \frac{h}{2}A(E_n y_n + y_{n+1}) \quad (2.2.28)$$

which is tested to be only first order accurate in time. Further investigations on exponential integrators based on explicit multi-step formulae, combining the ideas of the present paper to the results of [10] will give rise to a systematic order theory for methods like (2.2.27) and will be addressed elsewhere.

The incompressibility constraint of the Navier-Stokes equations leads to further complications in the design of high order numerical integrators, the reason being that after discretization in space these equations are not ordinary differential equations, but index 2 differential algebraic equations. For a discussion on these issues see for example [23].

2.2.3 Methods of order 2 and 3

The preliminary analysis of the order conditions reported in the previous section leads to the following conclusions:

- Any couple of classical RK methods of order 1 gives a new method of order 1.
- A couple of partitioned RK methods of order 2 gives a new method of order 2 .
- If we take a pair of PRK of order 3 (explicit + implicit) and construct a commutator-free method out of the explicit method in such a way that

$$\hat{b}_k = \sum_{l=1}^J \beta_l^k, \quad \hat{a}_{k,j} = \sum_{l=1}^J \alpha_{k,l}^j, \quad (2.2.29)$$

the resulting method satisfies the conditions for order 3 for the new class of methods.

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

We will use these observations to obtain methods up to and including order three in the new class of methods proposed in this paper. Our starting point are the IMEX methods of [1] and other partitioned Runge-Kutta methods reported in [13]. In all these methods $b_i = \hat{b}_i$ and $c_i = \hat{c}_i$ for $i = 1, \dots, s$.

In particular it is advantageous to consider methods requiring a minimal number of exponentials per stage, i.e. minimal J [8]. It is possible to obtain methods of order one and two with $J = 1$. For order three J must be at least 2. This implies that the methods of order one and two proposed in [1] can be used directly to obtain methods of the same order within the new class, while for order three we need to split up the coefficients of the explicit method according to (2.2.29) and also consider the extra (commutator-free) order condition from Table 2.5 (first and second row, coinciding when we assume $c_k = \hat{c}_k$).

From the first two conditions of order three in Table 2.5, setting $J = 2$ and using the conditions of order 1 and 2 one obtains

$$\sum_{k=1}^s (\beta_2^k + 2\beta_1^k c_k) = \frac{2}{3}. \quad (2.2.30)$$

Example 2.2.3. The couple of methods with tableaux

$$\begin{array}{c|cc} 0 & & \\ \frac{1}{2} & 0 & \frac{1}{2} \\ \hline & 0 & 1 \end{array}, \quad \begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & 1 \end{array},$$

[1], gives rise to the new method of order two

$$\begin{aligned} \varphi_{\frac{1}{2}} &= \exp\left(\frac{h}{2}C(y_0)\right), & Y_{\frac{1}{2}} &= \varphi_{\frac{1}{2}}y_0 + \frac{h}{2}AY_{\frac{1}{2}} \\ \varphi_1 &= \exp(hC(Y_{\frac{1}{2}})), & y_1 &= \varphi_1y_0 + h\varphi_1\varphi_{\frac{1}{2}}^{-1}AY_{\frac{1}{2}}. \end{aligned}$$

The corresponding transport-diffusion algorithm for convection-diffusion problems of type (2.1.1) has the form

$$\begin{aligned} \frac{D\tilde{u}_0}{Dt} &= 0, \quad \tilde{u}_0(x, 0) = u_0(x), \quad \text{on } [0, \frac{h}{2}], \\ \mathbf{V}(x) &= u_0(x), \\ \tilde{u}_0(x) &= \tilde{u}_0(x, \frac{h}{2}), \\ u_{\frac{1}{2}} &= \tilde{u}_0 + \frac{h}{2}\nu\nabla^2 u_{\frac{1}{2}} \\ \frac{D\tilde{u}_{\frac{1}{2}}}{Dt} &= 0, \quad \tilde{u}_{\frac{1}{2}}(x, 0) = \nu\nabla^2 u_{\frac{1}{2}}(x), \quad \text{on } [-\frac{h}{2}, 0], \\ \mathbf{V}(x) &= u_0(x), \\ u_{\frac{1}{2}}(x) &= \tilde{u}_{\frac{1}{2}}(x, -\frac{h}{2}), \\ \frac{D\tilde{u}_1}{Dt} &= 0, \quad \tilde{u}_1(x, 0) = u_0(x) + hu_{\frac{1}{2}}(x), \quad \text{on } [0, h], \end{aligned} \quad (2.2.31)$$

$$\begin{aligned} \mathbf{V}(x) &= u_{\frac{1}{2}}(x), \\ u_1(x) &= \tilde{u}_1(x, h). \end{aligned}$$

Example 2.2.4. In the next example we consider a second order IMEX method from [1], we implement this method in the numerical experiments as an IMEX or a DIRK commutator-free method obtaining in both cases order 2:

$$\begin{array}{c|ccc} 0 & & & \\ \gamma & \gamma & & \\ 1 & \delta & 1 - \delta & \\ \hline & 0 & 1 - \gamma & \gamma \end{array}, \quad \begin{array}{c|ccc} 0 & 0 & & \\ \gamma & 0 & \gamma & \\ 1 & 0 & 1 - \gamma & \gamma \\ \hline & 0 & 1 - \gamma & \gamma \end{array}$$

with $\gamma = \frac{2-\sqrt{2}}{2}$ and $\delta = \frac{-2\sqrt{2}}{3}$. The implicit method is stiffly accurate and L-stable.⁴

Example 2.2.5. For obtaining order 3 we consider as starting point the partitioned Runge-Kutta methods proposed by Griepentrog [13],

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}, \quad \begin{array}{c|ccc} 0 & 0 & & \\ \frac{1}{2} & -\frac{\beta}{2} & \frac{1+\beta}{2} & \\ 1 & \frac{3+5\beta}{2} & -(1+3\beta) & \frac{1+\beta}{2} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}.$$

The implicit method on the right is A-stable for $\beta = \sqrt{3}/3$. Starting from the explicit Runge-Kutta method on the left we can derive a commutator free method of order 3 using (2.2.29) and (2.2.30), we obtain

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{12} & \frac{1}{3} & -\frac{1}{4} \\ \hline & \frac{1}{12} & \frac{1}{3} & \frac{5}{12} \end{array},$$

this method coupled with the implicit method gives approximations of order three for (2.2.6).

Example 2.2.6. In the next example we have considered a third order IMEX method from [1] as a starting point and, using (2.2.29) and (2.2.30), we have obtained the following pair of Commutator-Free DIRK methods of order 3:

⁴Stiffly accurate Runge-Kutta methods, i.e. those whose last stage coincides with the numerical update, are particularly suited for index 1 differential algebraic equations.

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

$$\begin{array}{c|ccc}
 0 & & & \\
 \gamma & \gamma & & \\
 1-\gamma & \gamma-1 & 2(1-\gamma) & \\
 \hline
 & 0 & \frac{1}{2}-\varphi & \frac{1}{2}+\varphi \\
 \hline
 & 0 & \varphi & -\varphi
 \end{array}, \quad
 \begin{array}{c|ccc}
 0 & 0 & & \\
 \gamma & 0 & \gamma & \\
 1-\gamma & 0 & 1-2\gamma & \gamma \\
 \hline
 & 0 & \frac{1}{2} & \frac{1}{2}
 \end{array}$$

with $\gamma = \frac{3+\sqrt{3}}{6}$ and $\varphi = \frac{1}{6(2\gamma-1)}$.

Alternatively one could have considered the following commutator-free method:

$$\begin{array}{c|cccc}
 0 & & & & \\
 \gamma & \gamma & & & \\
 1-\gamma & \gamma-1 & 2(1-\gamma) & 0 & \\
 \hline
 & \alpha & \beta & \sigma & \\
 \hline
 & -\alpha & \frac{1}{2}-\beta & \frac{1}{2}-\sigma &
 \end{array}$$

where $\sigma = (\alpha + \beta(1 - 2\gamma) - \frac{1}{3})/(1 - 2\gamma)$, $\alpha = 1/2$, $\beta = 1/6$, and the same value of γ as above.

We have not found substantial difference in the numerical experiments between these two explicit commutator-free methods.

2.3 Numerical tests

We consider the short names DIRK for Diagonally Implicit Runge-Kutta methods, CF for Commutator-free method, IMEX for Implicit-Explicit, and SL for Semi-Lagrangian, and combinations of these. The methods considered in the numerical experiments are

- IMEX1 based on implicit and explicit Euler,
- DIRK-CF1 transport diffusion algorithm (by Pironneau),
- IMEX2 based on the midpoint, example (2.2.3),
- DIRK-CF2 based on the midpoint, example (2.2.3),
- IMEX2L IMEX based on an L-stable implicit DIRK method, example (2.2.4),
- DIRK-CF2L partitioned RK commutator-free method with L-stable DIRK method, example (2.2.4),
- IMEX3G partitioned RK method due to Griepentrog, example (2.2.5),
- DIRK-CF3G partitioned RK commutator-free method due to Griepentrog, example (2.2.5),
- IMEX3 IMEX method, example (2.2.6),
- DIRK-CF3 partitioned RK commutator-free method, example (2.2.6).

The numbers refer always to the order of the method. All the DIRK-CF methods have a semi-Lagrangian counterpart which we denote with SL1, SL2, SL2L, SL3G and SL3 respectively. In the semi-Lagrangian methods the exponentials of the convection operator are computed by approximating the characteristics of the corresponding pure convection problem and using interpolation. See the next section for an example and for further details on the approximation of the exact pure convection flows.

In the methods denoted by DIRK-CF the exponentials are simply matrix exponentials and if not specified otherwise they are computed by using the built-in MATLAB function `expm`.

2.3.1 Exponentials

We firstly consider a pure convection problem on the square with Dirichlet boundary conditions and with a convecting vector field tangent to the boundary of the square, i.e.

$$u_t(x, y, t) = V(x, y) \cdot \nabla u(x, y, t), \quad \mathbf{V}(x, y) = \begin{bmatrix} \pi \sin(\pi x) \cos(\pi y) \\ -\pi \cos(\pi x) \sin(\pi y) \end{bmatrix},$$

$(x, y) \in [0, 1] \times [0, 1]$, $t > 0$. The initial condition is $u_0(x, y) = u(x, y, 0) = e^{(-45(y-\frac{3}{4})^2 - 15(x-\frac{1}{2})^2)}$. We consider a discretization grid in space based on the Gauss-Lobatto-Legendre (GLL) points. We seek the approximate solution on the nodes of the grid at time $t = 1.2$.

This problem corresponds to the computation of a single exponential of the convection operator. The purpose of this experiment is to illustrate and propose two good strategies for the implementation of the exponentials of the methods presented in this paper.

The first approach consists in using an integration method for ordinary differential equations for computing fairly accurately the characteristics

$$\frac{dX}{d\tau} = \mathbf{V}(X(\tau)), \quad X(t_n + h) = \mathbf{x}, \quad \tau \in [t_n, t_n + h] \quad (2.3.1)$$

where \mathbf{x} denotes any point of \mathcal{G} the GLL discretization grid. We want to evaluate $u_n(\mathbf{y})$ at $\mathbf{y} = X(t_n)$ (the output of the ODE (2.3.1)) and, since $u_n(\mathbf{y})$ is explicitly known only for $\mathbf{y} \in \mathcal{G}$ (the grid points), we extend it by interpolation on the GLL nodes to the rest of the domain. We obtain

$$u(\mathbf{x}, t_n + h) \approx u_{n+1}(\mathbf{x}) := \sum_{\tilde{\mathbf{x}} \in \mathcal{G}} u_n(\tilde{\mathbf{x}}) l_{\tilde{\mathbf{x}}}(X(t_n))$$

where $l_{\tilde{\mathbf{x}}}$ is an element of the tensor product basis of Lagrange basis functions on the GLL nodes. See also [6, 11, 22] for similar approaches. In the experiment the `ode23` MATLAB routine with tolerances set at 10^{-3} was used to compute the characteristics.

Alternatively, using a spectral Galerkin approach based on the GLL points, one can construct the matrix C discretization of the convection operator, and compute the exponentials of the discrete operator using a restarted (preconditioned) Krylov subspace

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

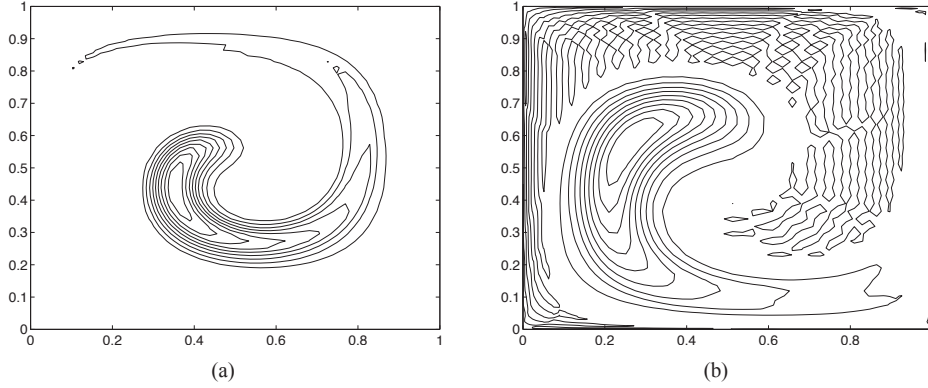


Figure 2.1: Pure convection with spectral methods in two dimensions. Approximation of the exponential of the convection operator. Semi-Lagrangian versus Eulerian method: **(a)** approximation of the characteristics trajectories with an accurate time-stepping procedure and interpolation of the initial condition; **(b)** semidiscretization in space and accurate Krylov subspace solution of the linear system of ODEs in time.

method for the approximation of the matrix exponential. The system of ODEs arising from the space discretization (using spectral Galerkin discretization of the convection operator) is

$$\dot{y} = Cy, \quad y(0) = y_0,$$

and each component of y_0 corresponds to $u_0(\tilde{x})$ for some $\tilde{x} \in \mathcal{G}$. One considers

$$y_{n+1} = \|y_n\|_2 V_k e^{hH_k} \mathbf{e}_1,$$

with

$$\text{span}(V_k) = \text{span}\{y_n, Cy_n, \dots, C^{k-1}y_n\}, \quad H_k := V_k^T C V_k,$$

$\text{span}(V_k)$ is the column range of V_k which is constructed to have orthonormal columns (i.e. $V_k^T V_k = I_k$ the $k \times k$ identity matrix) and $\mathbf{e}_1 \in \mathbf{R}^k$ is the first canonical vector. For details on the restarting procedure see [9], where also some cases of less trivial boundary conditions are covered. For preconditioning techniques for this method see [16]. We refer to this approach as Eulerian approach.

In both cases $n = 0$, $t_0 = 0$ and $h = 1.2$.

The results are shown in Figure 2.1. The semi-Lagrangian method does not produce any substantial numerical dispersion compared to the Eulerian approach, but, in our implementation, it does require more computational time. On the other hand the only possibility for improving the performance of the Eulerian method is to increase the number of points in the spatial discretization and this could cause slower convergence of the Krylov subspace method.

2.3.2 Korteweg de Vries equation (KdV)

In this and in the next subsection we consider two nonlinear test problems the KdV equation and the viscous Burgers' equation. Both the models involve a parameter ν and they coincide with the inviscid Burgers' equation when $\nu = 0$. In the limit as $\nu \rightarrow 0$ the solutions of these two equations have completely different behavior. This fact strongly affects the numerical discretizations.

We consider the KdV equation

$$u_t + uu_x = \nu u_{xxx}, \quad x \in [0, 2\pi], \quad t \in [0, 1]$$

with periodic boundary conditions, discretized by

$$U_t = D(U.^2) + AU$$

where D and A are obtained applying central differences for the first and third derivative in space on a uniform grid, with periodic boundary conditions, $U_i \approx u(x_i, t)$, $U.^2 = \text{diag}(U)U$ denotes the component-wise square of U . The operator $C(U) := D\text{diag}(U)$. The spatial discretization is such that $\Delta x = 2\pi/31$. In this experiment we verify that the order of the various time-integrators predicted by the theory is observed also in practice. We consider the following range of time-steps $h = 1/(2^4), \dots, 1/(2^9)$ and the initial condition $u_0 = 1.5 + \cos(x)$. The results are reported in Figure 2.2 (a), ($\nu = 0.05$).

This test problem was proposed in [18] where a study of the performance of numerical discretization of the KdV equation for different values of ν was considered. The solutions of the PDE become more and more oscillatory as ν decreases, requiring the use of tiny step-sizes in time and space for the numerical discretizations. See also [12] for a study of the small dispersion limit of the KdV equation. For small ν the use of up-winding techniques for the KdV equation is not recommended [18], in fact such techniques introduce artificial dissipation and fail to reproduce the oscillatory behavior of the solution.

The DIRK-CF methods are not structure preserving in time (symplectic, symmetric or energy-preserving methods), but in our experiments they perform well compared to their IMEX counterparts, and in general they might be cheaper to implement compared to most structure preserving integrators. The amount of artificial dissipation introduced by the proposed discretization of the KdV equation (central differences and DIRK-CF) is of the order of the time integration and therefore small for high order DIRK-CF methods. We think is possible to construct structure preserving methods belonging to the class of methods presented in this paper, for similar ideas in the context of exponential integrators see [5].

In this experiment the IMEX methods of order 2 and 3 produce a bigger error compared to the corresponding DIRK-CF methods. In Figure 2.2 (b) ($\nu = 0.01$) the numerical solutions obtained by the IMEX3 method and the DIRK-CF3 method are compared to the reference solution (denoted *exact* in the figure). The reference solution is computed with the same discretization in space ($\Delta x = 2\pi/255$). The time-integration is carried out by the MATLAB function `ode15s` (with tolerances 10^{-10}). The time interval is $[0, 4]$

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

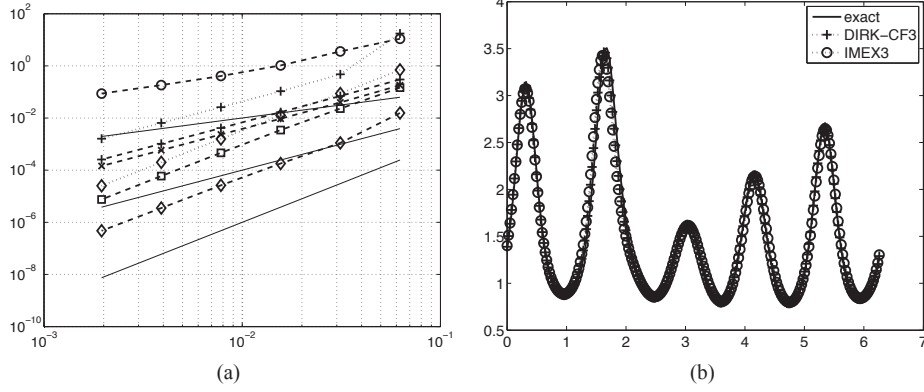


Figure 2.2: KdV equation: space discretization by central differences. **(a)** Order test: global error at $t = 1$, for different values of the time step h . $\nu = 0.05$, $x \in [0, 2\pi]$, $\Delta x = \frac{2\pi}{31}$. Dashed line DIRK-CF methods, dotted line IMEX methods, solid line: reference lines for order 1, 2 and 3. Symbols: (o) order 1; (x) order 2, (+) order 2 of type L; diamonds (\diamond) order 3 and squares (\square) order 3 type G. **(b)** Comparison of IMEX3 and DIRK-CF3 for the KdV equation: numerical solution at $t = 4$ as a function of $x \in [0, 2\pi]$, $\nu = 0.01$, $\Delta x = \frac{2\pi}{255}$, $h = \frac{1}{2^5}$ for DIRK-CF3 and $h = \frac{1}{2^6}$ for IMEX3.

and the time-step is $h = \frac{1}{2^6}$ for IMEX3 and $h = \frac{1}{2^5}$ for DIRK-CF3. Doubling the size of the time-step the IMEX3 method becomes unstable. In general the DIRK-CF3 method performs better than the corresponding IMEX3 method in this experiment also if we consider different values of ν or we increase the number of points in the spatial discretization.

2.3.3 1D Viscous Burgers' equation

In Figure 2.3, and 2.4 we consider

$$u_t + uu_x = \nu \nabla^2 u,$$

on $[0, 1]$ with homogeneous Dirichlet BCs, integrated on $[0, 1]$. We plot the relative error in the ∞ -norm versus the viscosity. The error is computed as follows:

$$\frac{\|U - U_{ref}\|_{\infty}}{\|U_{ref}\|_{\infty}}.$$

We discretize in space with a step $\Delta x = 1/81$ and in time with $h = 1.8\Delta x$. The reference solution, U_{ref} , is obtained discretizing in space with $\tilde{\Delta x} = 1/2592$ while the time integration of the discrete problem is performed in MATLAB by the built-in function `ode15s` with absolute and relative tolerances 10^{-8} . For small viscosity values the

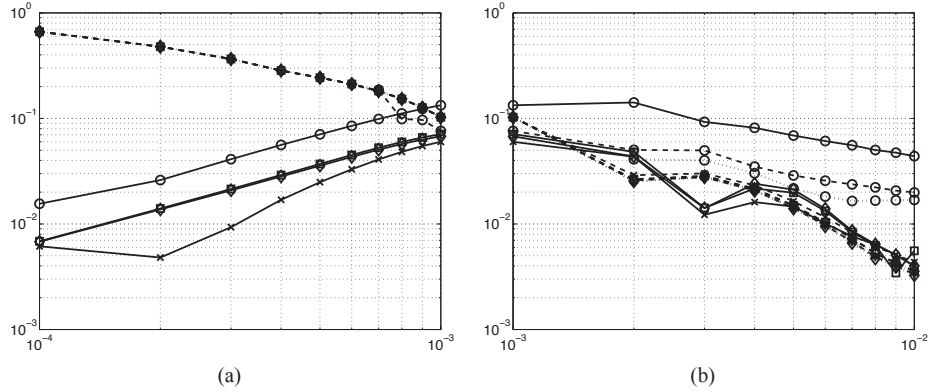


Figure 2.3: $u(x, 0) = \sin(\pi x)$, $\Delta x = 1/81$, $t = 2$, $h = 1.8\Delta x$: viscosity ν on the x -axis relative error y -axis, for IMEX, DIRK-CF and SL (semi-Lagrangian) methods with cubic spline interpolation, (a) $\nu \in [0.0001, 0.001]$ and (b) $\nu \in [0.001, 0.01]$. Dashed line DIRK-CF methods, dotted line IMEX methods, solid line SL methods. Symbols: (o) order 1; (x) order 2, (+) order 2 of type L; diamonds order 3 and squares order 3 type G. In these experiments the characteristic velocity $U \leq 1$ the Peclet number is $Pe \leq \frac{1}{81\nu}$ and the Courant number is 1.8.

semi-Lagrangian (SL) methods outperform their DIRK-CF and IMEX counterparts. The relative error of the Eulerian methods increases with decreasing viscosity (see Figure 2.3 (a)) some methods break down. Remarkably the second order semi-Lagrangian method performs even better than the two third order methods. Figure 2.3 considers cubic spline interpolation for the convecting vector field (using the routine `spline` of MATLAB) and the evaluation of the initial solution at the departure points of the characteristics, while in Figure 2.4 we used the built in MATLAB function `pchip`⁵ performing piecewise cubic, monotonic interpolation. In these two experiments the characteristic velocity $U \leq 1$ the Peclet number is $Pe = \frac{U\Delta x}{\nu} \leq \frac{1}{81\nu}$ and the Courant number is

$$\frac{h \max_{x \in [0,1]} |u(x)|}{\Delta x} = 1.8.$$

The results are comparable in the two figures. The choice of the interpolation method in the implementation of semi-Lagrangian methods is crucial for optimizing the performance at low viscosities, but this issue will not be discussed in detail in this paper.

⁵For more details about these MATLAB functions we refer to the MATLAB online helpdesk.

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

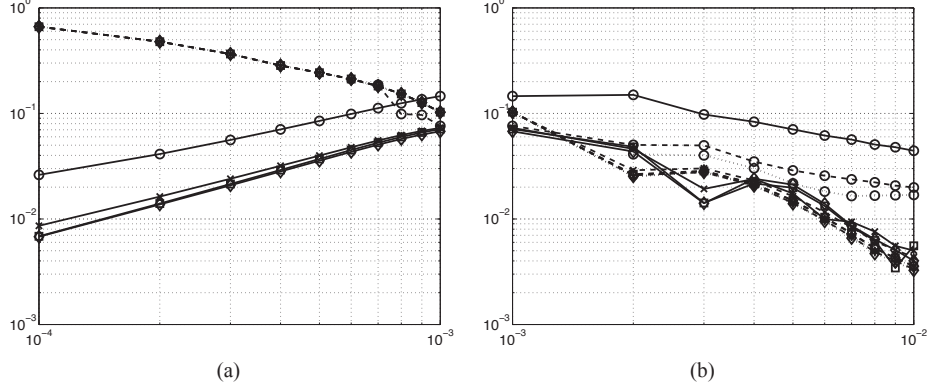


Figure 2.4: $u(x, 0) = \sin(\pi x)$, $\Delta x = 1/81$, $t = 2$, $h = 1.8\Delta x$: viscosity ν on the x -axis relative error y -axis, for IMEX, DIRK-CF and SL methods with piecewise cubic monotonic interpolation, **(a)** $\nu \in [0.0001, 0.001]$ and **(b)** $\nu \in [0.001, 0.01]$. Dashed line DIRK-CF methods, dotted line IMEX methods, solid line SL methods. Symbols: (o) order 1; (x) order 2, (+) order 2 of type L; diamonds order 3 and squares order 3 type G. In these experiments the characteristic velocity $U \leq 1$ the Peclet number is $Pe \leq \frac{1}{81\nu}$ and the Courant number is 1.8.

2.3.4 Linear convection and convection-diffusion in 2D

In the next experiment we consider linear convection-diffusion on the square $\Omega = [-1, 1]^2$ with viscosity $\nu = 10^{-4}$. We compare the performance of the Eulerian DIRK-CF3 method and of its semi-Lagrangian version SL3 integrating on the time interval $[0, 2\pi]$ using 40 time-steps (i.e. with step-size $h = 0.1571$). The same test case has been considered in [22].

The initial data is

$$u(x, y, 0) = e^{-[(x-x_0)^2 + (y-y_0)^2]/2\lambda^2},$$

with $(x_0, y_0) = (-0.5, 0)$, $\lambda = 1/8$. The solution of this PDE is

$$u(x, y, t) = \frac{\lambda^2}{\lambda^2 + 2\nu t} e^{-[\hat{x}^2 + \hat{y}^2]/2(\lambda^2 + 2\nu t)},$$

where $\hat{x} = x - x_0 \cos t - y_0 \sin t$, $\hat{y} = y + x_0 \sin t - y_0 \cos t$. The convecting vector field is

$$\mathbf{V}(x, y) = \begin{bmatrix} y \\ -x \end{bmatrix}.$$

We use a spectral Galerkin discretization in space based on polynomials of degree $p = 16$ on Gauss-Lobatto-Legendre nodes. In Figure 2.5 one can observe that the Eulerian approach has a tendency of introducing spurious oscillations in the numerical solution

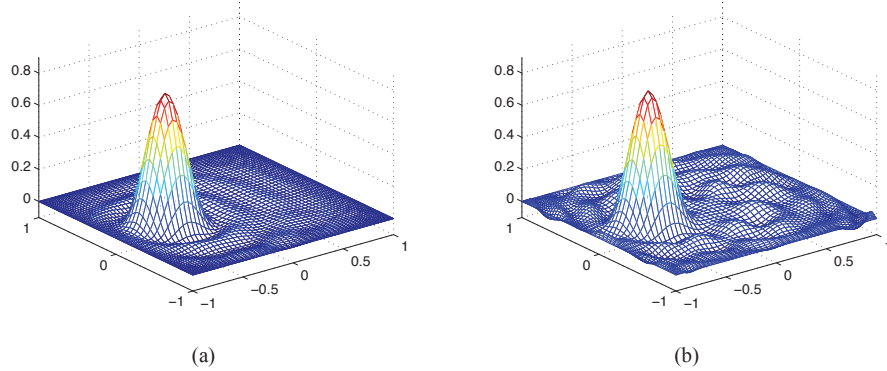


Figure 2.5: Linear convection-diffusion with spectral methods in two dimensions. Semi-Lagrangian versus Eulerian method: **(a)** SL3 integration in time, the diffusion is discretized on the GLL grid and the pure convection problems are approximated by the exact computation of the characteristics trajectories and the interpolation on the GLL nodes of the initial condition; **(b)** semi-discretization in space with a spectral Galerkin method and time integration performed with the DIRK-CF3 method, the exponentials of the convection operator are computed using the MATLAB built in function `expm`.

compared to the semi-Lagrangian method. The values of the relative error computed in the L_2 norm are of comparable size for the two approaches. We have an error of 0.0124 for the semi-Lagrangian method and of 0.0263 for its Eulerian counterpart. The maximum and minimum values of the numerical approximation are 0.8164 and -0.0239 for the SL3 and 0.8324 and -0.0500 for the DIRK-CF3. The maximum and minimum values of the exact solution are 0.9225 and 0.

Conclusions

We presented a new class of integrators for convection-dominated problems with a nonlinear convection term. The methods have good properties: they require the solution of only one symmetric linear system per stage, they have good linear stability properties and when implemented in a semi-Lagrangian fashion they do not produce numerical dispersion. The methods have been tested preliminary on the viscous Burgers' equation, the KdV equation and with a linear two dimensional convection-diffusion equation. The results are encouraging. A preliminary study of the order conditions has also been presented.

A more general setting for the methods would consider

$$\varphi_{i,j} := \exp\left(h \sum_k \alpha_{i,j,j}^k C(Y_k)\right) \cdots \exp\left(h \sum_k \alpha_{i,j,1}^k C(Y_k)\right),$$

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

in the formulae of section 2. We have not fully explored this possibility yet, but we believe that in such setting one should be able to design methods involving a smaller number of exponentials per stage compared to the present case.

We wish to acknowledge the anonymous referees for useful comments and discussions.

2.4 Appendix

2.4.1 Commutator-free methods for integration of ODEs on manifolds

A set of vector fields $\{\mathcal{E}_1, \dots, \mathcal{E}_d\}$ on a manifold \mathcal{M} of dimension $m \leq d$ is a set of frame vector fields if

$$T_x \mathcal{M} = \text{span}\{\mathcal{E}_1|_x, \dots, \mathcal{E}_d|_x\}, \quad \forall x \in \mathcal{M},$$

where $T_x \mathcal{M}$ is the tangent space to \mathcal{M} at the point x .

Given any vector field F on \mathcal{M} we have

$$F(y) = \sum_{i=1}^d f_i(y) \mathcal{E}_i(y).$$

We denote with F_p the vector field

$$F_p(x) = \sum_{i=1}^d f_i(p) \mathcal{E}_i(x)$$

we say that F_p is the vector field F frozen at the point p .

Given \mathcal{M} a manifold with a set of frame vector fields, we can define intrinsic Runge-Kutta like methods as follows:

Commutator-free method

$p = y_n$

for $r = 1 : s$ do

$$Y_r = \exp(\sum_{k=1}^s \alpha_{rJ}^k F_k) \cdots \exp(\sum_{k=1}^s \alpha_{r1}^k F_k)(p)$$

$$F_r = hF_{Y_r} = h \sum_{i=1}^d f_i(Y_r) \mathcal{E}_i$$

end

$$y_{n+1} = \exp(\sum_{k=1}^s \beta_J^k F_k) \cdots \exp(\sum_{k=1}^s \beta_1^k F_k)p$$

Here n counts the number of time steps and h is the step-size of integration. The integrator has s stages and parameters $\alpha_{rl}^k, \beta_l^k, r, k = 1, \dots, s$ and $l = 1, \dots, J$. Each new stage value is obtained as a composition of J exponentials (i.e. exact flows) of linear combinations of vector fields frozen at the previously computed stage values.

In the following tableaus we report the coefficients of a method of order 3 and a method of order 4. The method of order 3 requires the computation of one exponential of each internal stage value and the composition of two exponentials for updating the solution. In the order 4 method the first three stage values require one exponential each, while the fourth stage and the solution update require two exponentials. So in both cases $J = 2$.

0			
$\frac{1}{3}$	$\frac{1}{3}$		
$\frac{2}{3}$	0	$\frac{2}{3}$	
	$\frac{1}{3}$	0	0
	$-\frac{1}{12}$	0	$\frac{3}{4}$

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	$\frac{1}{2}$	0	0	
	$-\frac{1}{2}$	0	1	
	$\frac{1}{4}$	$\frac{1}{6}$	$\frac{1}{6}$	$-\frac{1}{12}$
	$-\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{4}$

If the manifold \mathcal{M} is the general linear group $\text{GL}(n)$ ⁶ then the typical vector field takes the form $C(U)U$ where $U \in \text{GL}(n)$ and $C(U) \in \mathfrak{gl}(n)$ ⁷. The vector field frozen at P has the form $C(P)U$ and the exponential of such a vector field is given by the solution of the differential equation

$$\dot{Y} = C(P)Y, \quad Y(0) = Y_0,$$

i.e. $Y(h) = \exp(hC(P))Y_0$. So $\exp(hC(P))$ is the matrix exponential of $C(P)$ and Y_0 is a suitable initial condition.

The commutator-free methods were presented in [8] and are a generalization of the methods proposed in [10]. The framework for constructing integration methods on manifolds presented in this section was first considered in [10].

References

- [1] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri, *Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations*, Appl. Numer. Math. **25** (1997), 151–167.
- [2] U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton, *Implicit-explicit methods for time-dependent partial differential equations*, SIAM J. Numer. Anal. **32** (1995), no. 3, 797–823.
- [3] M. J. Baines, *Moving finite elements*, Monographs on Numerical Analysis, The Clarendon Press Oxford University Press, New York, 1994, Oxford Science Publications. MR 1320498 (96c:65158)

⁶ $\text{GL}(n)$ is the set of invertible $n \times n$ matrices equipped with the matrix-matrix product as group product.

⁷ $\mathfrak{gl}(n)$ is the Lie algebra of $\text{GL}(n)$ and is the vector space of all $n \times n$ matrices.

Chapter 2. Semi-Lagrangian Runge-Kutta exponential integrators for convection-dominated problems

- [4] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral methods in fluid dynamics*, ed., , Springer-Verlag, Berlin, 1988.
- [5] D. Celledoni, E. Cohen and B. Owren, *Symmetric exponential integrators with an application to the cubic Schrödinger equation*, *Found. Comput. Math.* **8** (2008), no. 3, 303–317. MR 2413146 (2009f:65293)
- [6] E. Celledoni, *Eulerian and semi-Lagrangian commutator-free exponential integrators*, *CRM Proceedings and Lecture Notes* **39** (2005), 77–90.
- [7] E. Celledoni and B. K. Kometa, *Order conditions for semi-Lagrangian Runge-Kutta exponential integrators*, Preprint series: Numerics, 04/2009, Department of Mathematics, NTNU, Trondheim, Norway, <http://www.math.ntnu.no/preprint/numerics/N4-2009.pdf>, 2009.
- [8] E. Celledoni, A. Marthinsen, and B. Owren, *Commutator-free Lie group methods*, *FGCS* **19** (2003).
- [9] E. Celledoni and I. Moret, *A Krylov projection method for systems of ODEs*, *Appl. Numer. Math.* **24** (1997), no. 2-3, 365–378, Volterra centennial (Tempe, AZ, 1996). MR 1464736 (98g:65061)
- [10] P. E. Crouch and R. Grossman, *Numerical integration of ordinary differential equations on manifolds*, *J. Nonlinear Sci.* **3** (1993), no. 1, 1–33. MR 1216986 (94e:65069)
- [11] P.F. Fischer F.X. Giraldo, J.B. Perot, *A spectral element semi-lagrangian (SESL) method for the spherical shallow water equations*, *J. Comput. Phy.* **190**.
- [12] T. Grava and C. Klein, *Numerical solution of the small dispersion limit of Korteweg-de Vries and Whitham equations*, *Comm. Pure Appl. Math.* **60** (2007), no. 11, 1623–1664. MR 2349350 (2008h:35324)
- [13] E. Hairer and G. Wanner, *Solving ordinary differential equations. II*, second ed., Springer Series in Computational Mathematics, vol. 14, Springer-Verlag, Berlin, 1996, Stiff and differential-algebraic problems.
- [14] R.W. Hockney and J.W. Eastwood, *Computer simulations using particles*, ed., , McGraw-Hill, New York, 1981.
- [15] C. A. Kennedy and M. H. Carpenter, *Additive Runge-Kutta schemes for convection-diffusion-reaction equations*, *Appl. Numer. Math.* **44** (2003), no. 1-2, 139–181.
- [16] I. Moret and P. Novati, *RD-rational approximations of the matrix exponential*, *BIT* **44** (2004), no. 3, 595–615. MR 2106019 (2005h:65073)
- [17] B. Owren, *Order conditions for commutator-free Lie group methods*, *J. Phys. A* **39** (2006), no. 19, 5585–5599. MR 2220777 (2007e:37054)

-
- [18] P. Pietra and C. Pohl, *Weak limits of the quantum hydrodynamic model*, VLSI Design **9** (1999), 427–434.
- [19] O. Pironneau, *On the transport-diffusion algorithm and its applications to the Navier-Stokes equations*, Numer. Math. **38** (1981/82), no. 3, 309–332. MR 654100 (83d:65258)
- [20] M. D. Rees and K. W. Morton, *Moving point, particle, and free-Lagrange methods for convection-diffusion equations*, SIAM J. Sci. Statist. Comput. **12** (1991), no. 3, 547–572. MR 1093206 (92a:65253)
- [21] D. J. Tritton, *Physical fluid dynamics*, ed., , Van Nostrand Reinhold Co., Ltd., London, 1977.
- [22] D. Xiu and G. E. Karniadakis, *A semi-Lagrangian high-order method for Navier-Stokes equations*, J. Comput. Phys. **172** (2001), no. 2, 658–684. MR 1857617 (2002g:76077)
- [23] Z. Zheng and L. Petzold, *Runge-Kutta-Chebyshev projection method*, J. Comput. Phys. **219** (2006), no. 2, 976–991. MR 2274963 (2007m:65091)

Paper II

Order conditions for the semi-Lagrangian exponential integrators

Elena Celledoni and Bawfeh Kingsley Kometa

*Preprint series: Numerics, 2009-04, Department of
Mathematics, NTNU, Norway*

Chapter 1

Order conditions for the semi-Lagrangian exponential integrators

Abstract. Order conditions for the for the semi-Lagrangian Runge-Kutta exponential integrators [Celledoni and Kometa, J. Sc. Comput, 2009] are studied. In the latter paper it was shown that for a method of order 1 to 3, the order conditions are a subset of order conditions for the commutator-free Lie-group method (CF) and the parent partitioned Runge-Kutta method from which it is derived. Here we further show that for methods of order 4 in this class, there are extra coupling conditions. A new method is constructed and verified numerically to be of order 4.

1.1 Introduction

Semi-Lagrangian methods have been shown [1, 2, 3] to play an important role in the computation of flows of vector fields in exponential integrators designed for convection dominated problems of the convection-diffusion type. In this paper we examine some of the issues regarding the order conditions for the semi-Lagrangian exponential integrators, starting with a preliminary work by the authors in [3].

Suppose from the semi-discretization of a convection-diffusion model one obtains an ordinary differential equation (with initial data y_0) of the form

$$y_t = C(y)y + Ay, \quad y(0) = y_0, \quad (1.1.1)$$

with $y = y(t) \in \mathbb{R}^N$ for $t \in [0, T]$. The $N \times N$ matrices $C(y)$ and A represents the discrete convection and linear diffusion operators respectively.

The methods then take the following general format

Chapter 1. Order conditions for the semi-Lagrangian exponential integrators

for $i = 1 : s$ do

$$Y_i = \varphi_i y_n + h \sum_j a_{i,j} \varphi_{i,j} A Y_j,$$

$$\varphi_i = \exp\left(h \sum_k \alpha_{i,j}^k C(Y_k)\right) \dots \exp\left(h \sum_k \alpha_{i,1}^k C(Y_k)\right),$$

end

$$y_{n+1} = \varphi_{s+1} y_n + h \sum_i b_i \varphi_{s+1,i} A Y_i,$$

$$\varphi_{s+1} = \exp\left(h \sum_k \beta_j^k C(Y_k)\right) \dots \exp\left(h \sum_k \beta_1^k C(Y_k)\right),$$

where $\varphi_{i,j} = \varphi_i \varphi_j^{-1}$, $\{a_{i,j}, b_i\}$ are coefficients of a s -stage Runge-Kutta (RK) method and $\alpha_{i,l}^j$ and β_l^j are coefficients of a commutator-free (CF) Lie group method (studied in [4, 6]) defined on a RK method with coefficients $\{\hat{a}_{i,j}, \hat{b}_i\}$ such that

$$\hat{a}_{i,j} = \sum_{l=1}^J \alpha_{i,l}^j, \quad \hat{b}_i = \sum_{l=1}^J \beta_l^j. \quad (1.1.2)$$

Thus given a partition RK method with Butcher tableaux

$$\begin{array}{c|c} \mathbf{c} & \mathcal{A} \\ \hline & \mathbf{b} \end{array}, \quad \begin{array}{c|c} \hat{\mathbf{c}} & \hat{\mathcal{A}} \\ \hline & \hat{\mathbf{b}} \end{array}, \quad (1.1.3)$$

we treat the diffusion with the s -stage RK method $\{\mathcal{A}, \mathbf{b}, \mathbf{c}\}$ (preferably implicit) and the convection with a CF method based on the RK method $\{\hat{\mathcal{A}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}\}$.

Note: Here and in the rest of the literature, we shall write \sum_j (without explicit limits of summation) to actually mean $\sum_{j=1}^s$.

In the study of the under conditions we treat (for the sake of convenience) the numerical solution y_{n+1} as an extra state value

$$Y_{s+1} = \varphi_{s+1} y_n + h \sum_j a_j \varphi_{s+1,j} A Y_j, \quad a_{s+1,j} = b_j,$$

with

$$\varphi_{s+1} = \exp\left(h \sum_k \alpha_{s+1,j}^k C(Y_k)\right) \dots \exp\left(h \sum_k \alpha_{s+1,1}^k C(Y_k)\right), \quad \alpha_{s+1,l}^k = \beta_l^k$$

and $\varphi_{s+1,j} = \varphi_{s+1} \varphi_j^{-1}$.

1.2 Deriving the order conditions

A preliminary work on the order conditions had been carried out by the authors in [3]. However, for the sake of completion, we present the whole derivations in this paper.

Taking the q^{th} derivatives with respect to h of the exact solution to (1.1.1) and of the stage values of the numerical solution we obtain the recursive formulas

$$y^{(q)} = \sum_{k=0}^{q-1} \binom{q-1}{k} \frac{d^{q-1-k}}{dh^{q-1-k}} (C(y) + A)y^{(k)}, \quad (1.2.1)$$

$$Y_i^{(q)} = \varphi_i^{(q)} y_0 + q \sum_j a_{i,j} \sum_{k=0}^{q-1} \binom{q-1}{k} \varphi_{i,j}^{(q-1-k)} A Y_j^{(k)}. \quad (1.2.2)$$

Order conditions for order $p = 1, 2, 3, \dots$ are recursively from the equations

$$Y_{s+1}^{(q)}|_{h=0} = y^{(q)}|_{h=0}, \quad q = 0, 1, \dots, p. \quad (1.2.3)$$

We often will simplify higher order conditions using conditions of lower order whenever necessary. The computation of the derivatives of Y_i requires the use of φ_i and $\varphi_{i,j}$ and their derivatives.

Now let us consider the matrix-valued functions,

$$C_{i,J-l} := h \sum_k \alpha_{i,J-l}^k C(Y_k), \quad l = 0, 1, \dots, J-1,$$

and

$$\tilde{C}_{i,J-l} := -h \sum_k \alpha_{i,l+1}^k C(Y_k), \quad l = 0, 1, \dots, J-1$$

We denote by $B_{i,J-l}$ either of $C_{i,J-l}$ or $\tilde{C}_{i,J-l}$, for $l = 0, 1, \dots, J-1$, and consider

$$\psi_i(h) := \exp(B_{i,J}) \cdot \exp(B_{i,J-1}) \cdot \dots \cdot \exp(B_{i,1}).$$

Depending on the choice of $B_{i,J-l} = C_{i,J-l}$ or $\tilde{C}_{i,J-l}$, we have $\psi_i = \varphi_i$ or φ_i^{-1} , respectively. We will also make use of

$$\varphi_i^l(h) := \exp(B_{i,J}) \cdot \exp(B_{i,J-1}) \cdot \dots \cdot \exp(B_{i,J-l}).$$

We obtain¹

$$\dot{\psi}_i = \sum_{l=0}^{J-1} \text{Ad}_{\psi_i^l} \left(\text{dexp}_{B_{i,J-l}}(\dot{B}_{i,J-l}) \right) \cdot \psi_i.$$

So we can write

$$\dot{\psi}_i = S_i(h) \psi_i \text{ with } S_i(h) := \sum_{l=0}^{J-1} \text{Ad}_{\psi_i^l} \left(\text{dexp}_{B_{i,J-l}}(\dot{B}_{i,J-l}) \right),$$

¹We recall that $\text{dexp}_w(u) := \frac{e^z - 1}{z} \Big|_{z=\text{ad}_w} (u) = u + 1/2![w, u] + 1/3![w, [w, u]] + \dots$, $\text{ad}_w(u) := [w, u]$ (matrix commutator of w and u) and $\text{Ad}_\psi(u) := \psi u \psi^{-1}$.

Chapter 1. Order conditions for the semi-Lagrangian exponential integrators

and as a direct consequence we have

$$\psi_i^{(r)} = \sum_{k=0}^{r-1} \binom{r-1}{k} \left(\frac{d^{r-1-k}}{dh^{r-1-k}} S_i(h) \right) \psi_i^{(k)}. \quad (1.2.4)$$

Now we have the following proposition for finding the derivatives of $S_i(h)$:

Proposition 1.2.1. *Given that $Z^0 = Z^0(h)$ and $W = W(h)$ are two matrix-valued differentiable functions then*

$$\frac{d^r}{dh^r} \text{Ad}_W Z^0 = \text{Ad}_W Z^r, \quad (1.2.5)$$

with

$$Z^r = [W^{-1} \dot{W}, Z^{r-1}] + \dot{Z}^{r-1}. \quad (1.2.6)$$

The proof is by induction.

By differentiating from (1.2.6) we obtain

$$\dot{Z}^r = [\dot{W}^{-1} \dot{W} + W^{-1} \ddot{W}, Z^{r-1}] + [W^{-1} \dot{W}, \dot{Z}^{r-1}] + \ddot{Z}^{r-1}, \quad (1.2.7)$$

and using (1.2.6) and (1.2.7), assuming $W(0) = I$, we obtain

$$\left\{ \begin{array}{l} \frac{d}{dh} \text{Ad}_W Z^0|_{h=0} = Z^1(0) = [\dot{W}(0), Z^0(0)] + \dot{Z}^0(0) \\ \dot{Z}^1(0) = [-\dot{W}(0)^2 + \ddot{W}(0), Z^0(0)] + [\dot{W}(0), \dot{Z}^0(0)] + \ddot{Z}^0(0) \\ \frac{d^2}{dh^2} \text{Ad}_W Z^0|_{h=0} = Z^2(0) = [\dot{W}(0), Z^1(0)] + \dot{Z}^1(0) \\ \dot{Z}^2(0) = [-\dot{W}(0)^2 + \ddot{W}(0), Z^1(0)] + [\dot{W}(0), \dot{Z}^1(0)] + \ddot{Z}^1(0) \\ \frac{d^3}{dh^3} \text{Ad}_W Z^0|_{h=0} = Z^3(0) = [\dot{W}(0), Z^2(0)] + \dot{Z}^2(0) \\ \vdots \end{array} \right. \quad (1.2.8)$$

Further assuming that $Z^0 = \text{dexp}_{-B}(\dot{B})$ for some matrix-valued function $B = B(h)$, expanding the right-hand side and differentiating we obtain

$$\left\{ \begin{array}{l} Z^0(0) = \dot{B}(0) \\ \dot{Z}^0(0) = \ddot{B}(0) \\ \ddot{Z}^0(0) = \ddot{B}(0) - \frac{1}{2}[\dot{B}(0), [\dot{B}(0), \ddot{B}(0)]] \\ \ddot{\ddot{Z}}^0(0) = B^{(iv)}(0) - [\dot{B}(0), \ddot{B}(0)] + \frac{1}{2}[\dot{B}(0), \ddot{B}(0)] \\ \vdots \end{array} \right. \quad (1.2.9)$$

We can now obtain derivatives of S_i and ψ_i . By setting $W = \psi_i^l$ and $B = B_i^{J-l}$ we can calculate the derivatives of S_i using the steps in (1.2.5)-(1.2.9). We obtain

$$\left\{ \begin{array}{l} S_i(0) = \sum_{l=0}^{J-1} \dot{B}_i^{J-l}(0), \\ \frac{dS_i}{dh} \Big|_{h=0} = \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} ([\dot{B}_i^{J-r}(0), \dot{B}_i^{J-l}(0)] + \ddot{B}_i^{J-l}(0)), \\ \frac{d^2 S_i}{dh^2} \Big|_{h=0} = 2 \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} [\dot{B}_i^{J-r}(0), \ddot{B}_i^{J-l}(0)] + \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} [\ddot{B}_i^{J-r}(0), \dot{B}_i^{J-l}(0)] + \\ + \sum_{l=0}^{J-1} (\ddot{B}_i^{J-r}(0) - \frac{1}{2}[\dot{B}_i^{J-l}(0), \ddot{B}_i^{J-l}(0)]), \\ \vdots \end{array} \right. \quad (1.2.10)$$

Analogously the derivatives of $S_i^l := \sum_{r=0}^{J-l-1} \text{Ad}_{\psi_i^r} (\text{dexp}_{B_{i,J-r}}(\dot{B}_{i,J-r}))$ are obtained as in the forgoing formulae but substituting $J-1$ as upper index in the external summation with $J-l-1$. In Table 1.1 we report the values of the derivatives of φ_i and φ_j^{-1} at $h=0$, which are obtained from (1.2.4) and (1.2.10) by recursion, starting with $\psi_i(0) = I$. In Table 1.2 we report the values of the derivatives of $\varphi_{i,j}$ at $h=0$, which are obtained using Table 1.1 and the formula

$$\varphi_{i,j}^{(m)} = \sum_{r=0}^m \binom{m}{r} \varphi_i^{(m-r)} (\varphi_j^{-1})^{(r)}. \quad (1.2.11)$$

The derivatives of Y_i , reported in Table 1.3, are obtained using the results in tables 1.1 and 1.2, and the recursion formula (1.2.2), starting with $Y_i(0) = y_0$.

1.3 Order conditions for orders 1 – 3

We now present a detailed analysis for deriving the third order conditions.

From (1.2.4) we obtain that

$$\dot{\varphi}_i(0) = \sum_{l=0}^{J-1} \dot{B}_{i,J-l}(0) = \sum_{l=0}^{J-1} \dot{C}_{i,J-l}(0) \sum_{k=0}^{J-l-1} \sum_k \alpha_{i,J-l}^k C = \left(\sum_k \hat{a}_{i,k} \right) C. \quad (1.3.1)$$

Analogously one computes $\dot{\varphi}_j^{-1}(0)$. These expressions are reported in Table 1.1 and can be used to obtain

$$\dot{Y}_i|_{h=0} = \left(\sum_k \hat{a}_{i,k} \right) C y_0 + \left(\sum_k a_{i,k} \right) A y_0 = \hat{c}_i C y_0 + c_i A y_0, \quad (1.3.2)$$

Chapter 1. Order conditions for the semi-Lagrangian exponential integrators

see Table 1.3. Also from (1.2.1) we obtain

$$\dot{y}(0) = Cy_0 + Ay_0. \quad (1.3.3)$$

Imposing $\dot{Y}_{s+1}|_{h=0} = \dot{y}(0)$ we obtain the following order conditions for order 1,

$$\sum_k \hat{a}_{s+1,k} = 1, \quad \sum_k a_{s+1,k} = 1.$$

These correspond to requiring that the two RK methods (1.1.3) are consistent.

For order 2 from (1.2.2) we have that

$$\ddot{Y}_i|_{h=0} = \varphi_i^{(2)}|_{h=0}y_0 + 2 \sum_j a_{i,j} \left(\varphi_{i,j}^{(1)}(0) + \varphi_{i,j}(0)A\dot{Y}_j(0) \right)$$

with $\varphi_{i,j}(0) = I$ and $\varphi_{i,j}^{(1)}(0) = \dot{\varphi}_i(0) - \dot{\varphi}_j(0) = (\hat{c}_i - \hat{c}_j)C$. Using $\dot{Y}_j(0)$ from Table 1.3 we obtain

$$\ddot{Y}_i|_{h=0} = \varphi_i^{(2)}|_{h=0}y_0 + 2 \sum_j a_{i,j}((\hat{c}_i - \hat{c}_j)CAy_0 + \hat{c}_jACy_0 + c_jA^2y_0). \quad (1.3.4)$$

From (1.2.4) and (1.2.10) we obtain

$$\begin{aligned} \varphi_i^{(2)}|_{h=0} &= \frac{dS_i(h)}{dh} \varphi_i(h)|_{h=0} + (S_i(h)^2 \varphi_i(h))|_{h=0} \\ &= 2 \sum_k \hat{a}_{i,k} \hat{c}_k C'(Cy_0) + 2 \sum_k \hat{a}_{i,k} c_k C'(Ay_0) + (\sum_j \hat{a}_{i,j})^2 C^2, \end{aligned} \quad (1.3.5)$$

reported in Table 1.1. Substituting the results in (1.3.4) we obtain

$$\begin{aligned} \ddot{Y}_i|_{h=0} &= (2 \sum_k \hat{a}_{i,k} \hat{c}_k C'(Cy_0) + 2 \sum_k \hat{a}_{i,k} c_k C'(Ay_0) + c_i^2 C^2)y_0 + \\ &2 \sum_j a_{i,j}((\hat{c}_i - \hat{c}_j)CAy_0 + \hat{c}_jACy_0 + c_jA^2y_0), \end{aligned} \quad (1.3.6)$$

reported in Table 1.3. Using (1.2.1) and substituting for $\dot{y}(0)$ from (1.3.3) we obtain

$$y^{(2)}|_{h=0} = C'(\dot{y}(0))y_0 + (C + A)^2y_0 = C'((C + A)y_0)y_0 + (C + A)^2y_0, \quad (1.3.7)$$

where $C'(y)(w)$ is obtained by differentiating $C(y)$ such that

$$(C'(y)(w))_{i,j} := \sum_{k=1}^N \frac{\partial c_{i,j}}{\partial y^k}(y)w_k, \quad c_{i,j} = (C(y))_{i,j}, \quad y = [y^1, \dots, y^N]^T.$$

Taking $i = s + 1$ and matching coefficients in $\ddot{Y}_i|_{h=0}$ and $y^{(2)}|_{h=0}$ we obtain the four order conditions for order 2,

Table 1.1: Derivatives of φ_i and its inverse at $h = 0$, where $\overline{\sum} := \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} \sum_k \sum_m$.

q	$\varphi_i^{(q)}(0)$
0	I
1	$C\hat{c}_i$
2	$2 \sum_k \hat{a}_{i,k} C'(\hat{c}_k C y_0 + c_k A y_0) + \hat{c}_i^2 C^2$
3	$4 \overline{\sum} \alpha_{i,J-r}^k \alpha_{i,J-l}^m [C, C'(\hat{c}_m C + c_m A)] + 2 \overline{\sum} \alpha_{i,J-r}^k \alpha_{i,J-l}^m [C'(\hat{c}_k C + c_k A), C] +$ $- \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \sum_m \alpha_{i,J-l}^m [C, C'(\hat{c}_m C + c_m A)] +$ $3 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k C''(\hat{c}_k C + c_k A, \hat{c}_k C + c_k A) +$ $6 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \sum_j \hat{a}_{k,j} C'(C'(\hat{c}_j C + c_j A)) + 3 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \hat{c}_k^2 C'(C^2) +$ $6 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k (\hat{c}_k c_k - \sum_j a_{k,j} \hat{c}_j) C'(CA) +$ $6 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \sum_j a_{k,j} C'(A(\hat{c}_j C + c_j A)) +$ $4 \hat{c}_i \sum_k a_{i,k} (C'(\hat{c}_k C + c_k A)) C + 2 \hat{c}_i \sum_k a_{i,k} C(C'(\hat{c}_k C + c_k A)) + \hat{c}_i^3 C^3$
q	$(\varphi_j^{-1})^{(q)}(0)$
0	I
1	$-C\hat{c}_j$
2	$-2 \sum_k \hat{a}_{j,k} C'(\hat{c}_k C y_0 + c_k A y_0) + \hat{c}_j^2 C^2$
3	$4 \overline{\sum} \alpha_{j,r+1}^k \alpha_{i,r+1}^m [C, C'(\hat{c}_m C + c_m A)] + 2 \overline{\sum} \alpha_{j,r+1}^k \alpha_{i,r+1}^m [C'(\hat{c}_k C + c_k A), C] +$ $- \sum_{l=0}^{J-1} \sum_k \alpha_{j,l+1}^k \sum_r \alpha_{j,l+1}^r [C, C'(\hat{c}_r C + c_r A)] +$ $- 3 \sum_{l=0}^{J-1} \sum_k \alpha_{j,l+1}^k C''(\hat{c}_k C + c_k A, \hat{c}_k C + c_k A) +$ $- 6 \sum_{l=0}^{J-1} \sum_k \alpha_{j,l+1}^k \sum_j \hat{a}_{k,j} C'(C'(\hat{c}_j C + c_j A)) - 3 \sum_{l=0}^{J-1} \sum_k \alpha_{j,l+1}^k \hat{c}_k^2 C'(C^2) +$ $- 6 \sum_{l=0}^{J-1} \sum_k \alpha_{j,l+1}^k (\hat{c}_k c_k - \sum_j a_{k,j} \hat{c}_j) C'(CA) +$ $- 6 \sum_{l=0}^{J-1} \sum_k \alpha_{j,l+1}^k \sum_r a_{k,r} C'(A(\hat{c}_r C + c_r A)) +$ $4 \hat{c}_j \sum_k a_{i,k} (C'(\hat{c}_k C + c_k A)) C + 2 \hat{c}_j \sum_k a_{j,k} C(C'(\hat{c}_k C + c_k A)) + \hat{c}_j^3 C^3$

$$\sum_j \hat{a}_{s+1,j} \hat{c}_j = \frac{1}{2}, \quad \sum_j a_{s+1,j} c_j = \frac{1}{2},$$

$$\sum_j a_{s+1,j} \hat{c}_j = \frac{1}{2}, \quad \sum_j \hat{a}_{s+1,j} c_j = \frac{1}{2}.$$

Chapter 1. Order conditions for the semi-Lagrangian exponential integrators

Table 1.2: Derivatives of $\varphi_{i,j}$ at $h = 0$.

q	$\varphi_{i,j}^{(q)}(0)$
0	I
1	$(\hat{c}_i - \hat{c}_j)C$
2	$2 \sum_k (\hat{a}_{i,k} - \hat{a}_{j,k})C'(\hat{c}_k C + c_k A) + (\hat{c}_i - \hat{c}_j)^2 C^2$
3	$ \begin{aligned} & 4 \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} (\sum_k \alpha_{i,J-r}^k \sum_m \alpha_{i,J-l}^m - \sum_k \alpha_{i,r+1}^k \sum_m \alpha_{i,l+1}^m) [C, C'(\hat{c}_m C + c_m A)] \\ & + 2 \sum_{l=0}^{J-1} \sum_{r=0}^{J-l-1} (\sum_k \alpha_{i,J-r}^k \sum_m \alpha_{i,J-l}^m - \sum_k \alpha_{i,r+1}^k \sum_m \alpha_{i,l+1}^m) [C'(\hat{c}_k C + c_k A), C] \\ & - \sum_{l=0}^{J-1} (\sum_k \alpha_{i,J-l}^k \sum_m \alpha_{i,J-l}^m - \sum_k \alpha_{i,l+1}^k \sum_m \alpha_{i,l+1}^m) [C, C'(\hat{c}_m C + c_m A)] + \\ & 3 \sum_k (\hat{a}_{i,k} - \hat{a}_{j,k}) C''(\hat{c}_k C + c_k A, \hat{c}_k C + c_k A) + \\ & 6 \sum_k (\hat{a}_{i,k} \sum_m \hat{a}_{k,m} - \hat{a}_{j,k} \sum_m \hat{a}_{k,m}) C'(C'(\hat{c}_m C + c_m A)) + \\ & 3 \sum_k (\hat{a}_{i,k} - \hat{a}_{j,k}) \hat{c}_k^2 C'(C^2) + \\ & 6 \sum_k (\hat{a}_{i,k} - \hat{a}_{j,k}) \sum_m a_{k,m} C'(A(\hat{c}_m C + c_m A)) + \\ & 4 \sum_k (\hat{c}_i \hat{a}_{i,k} + \hat{c}_j \hat{a}_{j,k}) C'(\hat{c}_k C + c_k A) C + 2 \sum_k (\hat{c}_i \hat{a}_{i,k} + \hat{c}_j \hat{a}_{j,k}) C C'(\hat{c}_k C + c_k A) + \\ & (\hat{c}_i^3 - \hat{c}_j^3) C^3 + 3(\hat{c}_i \hat{c}_j^2 - \hat{c}_j \hat{c}_i^2) C^3 - 6 \hat{c}_j \sum_k \hat{a}_{i,k} C'(\hat{c}_k C + c_k A) C + \\ & - 6 \hat{c}_i \sum_k \hat{a}_{j,k} C C'(\hat{c}_k C + c_k A) \end{aligned} $

Note: The matrix-valued function $C = C(y)$ and its derivatives are linear with respect to y .

For order 3 we proceed as follows:

First from (1.2.1) we have

$$\begin{aligned}
y^{(3)}|_{h=0} &= C''(y_0)(\dot{y}(0), \dot{y}(0))y_0 + C'(y_0)(\ddot{y}(0))y_0 + (C(y_0) + A)^3 y_0 + \\
& 2C'(y_0)(\dot{y}(0))(C(y_0) + A)y_0 + (C(y_0) + A)C'(y_0)(\dot{y}(0))y_0,
\end{aligned} \tag{1.3.8}$$

where we have used $C''(y)(w, z)$ obtained by differentiating $C(y)$ such that

$$(C''(y))(w, z)_{i,j} := \sum_{k=1}^N \sum_{m=1}^N \frac{\partial^2 c_{i,j}}{\partial y^k \partial y^m}(y) w_k z_m, \quad c_{i,j} = (C(y))_{i,j}.$$

In short we will write C, C', C'', \dots for $C(y_0), C'(y_0), C''(y_0), \dots$ respectively. Substituting for $\dot{y}(0)$ and $\ddot{y}(0)$ from (1.3.3) and (1.3.7) respectively, we obtain

$$\begin{aligned}
y^{(3)}|_{h=0} &= C''((C + A)y_0, (C + A)y_0)y_0 + C'(C'((C + A)y_0)y_0 + (C + A)^2 y_0)y_0 + \\
& 2C'((C + A)y_0)(C + A)y_0 + (C + A)C'((C + A)y_0)y_0 + (C + A)^3 y_0.
\end{aligned} \tag{1.3.9}$$

Table 1.3: Derivatives of Y_i at $h = 0$.

q	$Y_i^{(q)}(0)$
0	y_0
1	$(\sum_j \hat{a}_{i,j})Cy_0 + (\sum_j a_{i,j})Ay_0$
2	$2 \sum_j \hat{a}_{i,j}C'(\hat{c}_jC + c_jA)y_0 + \hat{c}_i^2C^2y_0 +$ $2\hat{c}_i c_i CAy_0 - 2(\sum_j a_{i,j}\hat{c}_j)CAy_0 + 2 \sum_j a_{i,j}A(\hat{c}_jC + c_jA)y_0$
3	$4 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \alpha_{i,J-r}^k \sum_m \alpha_{i,J-l}^m [C, C'(\hat{c}_mC + c_mA)]y_0$ $+ 2 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \alpha_{i,J-r}^k \sum_m \alpha_{i,J-l}^m [C'(\hat{c}_kC + c_kA), C]y_0$ $- \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \sum_r \alpha_{i,J-l}^r [C, C'(\hat{c}_rC + c_rA)]y_0 +$ $3 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k C''(\hat{c}_kC + c_kA, \hat{c}_kC + c_kA)y_0$ $+ 6 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \sum_j \hat{a}_{k,j}C'(C'(\hat{c}_jC + c_jA))y_0 +$ $3 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \hat{c}_k^2 C'(C^2)y_0 +$ $6 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k (\hat{c}_k c_k - \sum_j a_{k,j}\hat{c}_j)C'(CA)y_0 +$ $6 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \sum_j a_{k,j}C'(A(\hat{c}_jC + c_jA))y_0 +$ $4\hat{c}_i \sum_k a_{i,k}(C'(\hat{c}_kC + c_kA))Cy_0 + 2\hat{c}_i \sum_k a_{i,k}C(C'(\hat{c}_kC + c_kA))y_0 + \hat{c}_i^3 C^3 y_0$ $6 \sum_k \hat{a}_{i,k}(\sum_j a_{i,j})C'(\hat{c}_kC + c_kA)Ay_0 + 3(\sum_k \hat{a}_{i,k})^2(\sum_j a_{i,j})C^2 Ay_0 +$ $6 \sum_k \hat{a}_{i,k}(-\sum_j a_{i,j}\hat{c}_j C^2 Ay_0 + \sum_j a_{i,j}CA(\hat{c}_jC + c_jA)y_0 + 3 \sum_j a_{i,j}\hat{c}_j^2 C^2 Ay_0 -$ $6 \sum_j a_{i,j} \sum_k \hat{a}_{j,k}C'(\hat{c}_kC + c_kA)Ay_0 - 6 \sum_j a_{i,j}\hat{c}_j CA(\hat{c}_jC + c_jA)y_0 +$ $6 \sum_j a_{i,j} \sum_m \hat{a}_{j,m}AC'(\hat{c}_mC + c_mA)y_0 + 3 \sum_j a_{i,j}\hat{c}_j^2 AC^2 y_0 +$ $6 \sum_j a_{i,j}(\hat{c}_j c_j - \sum_m a_{j,m}\hat{c}_m)ACAy_0 + 6 \sum_j a_{i,j} \sum_m a_{j,m}A^2(\hat{c}_mC + c_mA)y_0$

We now consider the third derivative of the numerical solution. From (1.2.2) we obtain

$$\begin{aligned}
Y_i^{(3)}\Big|_{h=0} &= \varphi_i^{(3)}\Big|_{h=0} y_0 + 3 \sum_j a_{i,j} \varphi_i^{(2)}\Big|_{h=0} Ay_0 + \\
&6 \sum_j a_{i,j} \varphi_i^{(1)}\Big|_{h=0} A\dot{Y}_j\Big|_{h=0} + 3 \sum_j a_{i,j} A \ddot{Y}_j\Big|_{h=0}.
\end{aligned} \tag{1.3.10}$$

We need to find $\varphi_i^{(3)}\Big|_{h=0}$ via (1.2.4) using the expressions for $\varphi_i^{(2)}\Big|_{h=0}$ and $\varphi_i^{(1)}\Big|_{h=0}$ which have already been found and reported in Table 1.1. Using earlier row entries of Table 1.1 and (1.2.11) we also compute $\varphi_{i,j}^{(2)} = \varphi_i(0) + 2\ddot{\varphi}_i(0)\varphi_j^{-1}(0) + \varphi_j^{-1}(0)$, reported in Table

Chapter 1. Order conditions for the semi-Lagrangian exponential integrators

1.2.

From (1.2.4) it follows that

$$\varphi_i^{(3)} \Big|_{h=0} = \frac{d^2 S_i}{dh^2} \Big|_{h=0} + 2 \frac{dS_i}{dh} \Big|_{h=0} \varphi_i^{(1)} \Big|_{h=0} + S_i \Big|_{h=0} \varphi_i^{(2)} \Big|_{h=0}. \quad (1.3.11)$$

We obtain

$$\begin{aligned} \varphi_i^{(3)} \Big|_{h=0} &= \frac{d^2 S_i}{dh^2} \Big|_{h=0} + \\ &2(2 \sum_k \hat{a}_{i,k} \hat{c}_k C'(Cy_0) + 2 \sum_k \hat{a}_{i,k} c_k C'(Ay_0)) \sum_k \hat{a}_{i,k} C + \\ &2 \sum_k \hat{a}_{i,k} C (\sum_k \hat{a}_{i,k} \hat{c}_k C'(Cy_0) + \sum_k \hat{a}_{i,k} c_k C'(Ay_0)) + \sum_k \hat{a}_{i,k} C (\sum_j \hat{a}_{i,j})^2 C^2. \end{aligned} \quad (1.3.12)$$

We have

$$\begin{aligned} \dot{C}_{i,J-l}(0) &= \sum_k \alpha_{i,J-l}^k C, \\ \ddot{C}_{i,J-l}(0) &= 2 \sum_k \alpha_{i,J-l}^k C'(\hat{c}_k C + c_k A), \\ \dddot{C}_{i,J-l}(0) &= 3 \sum_k \alpha_{i,J-l}^k C''(\hat{c}_k C + c_k, \hat{c}_k C + c_k) + C'(\dot{Y}_k(0)). \end{aligned} \quad (1.3.13)$$

We use (1.2.10) to find $\frac{d^2 S_i}{dh^2} \Big|_{h=0}$, setting $B_{i,J-l} = C_{i,J-l}$, and using the derivatives computed in (1.3.13) with $\dot{Y}_k(0)$ from Table 1.3.

Finally we get

$$\begin{aligned} \frac{d^2 S_i}{dh^2} \Big|_{h=0} &= \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_{k,m} \alpha_{i,J-l}^k \alpha_{i,J-l}^m (4[C, C'(\hat{c}_m C + c_m A)] + 2[C'(\hat{c}_k C + c_k A), C]) + \\ &3 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k C''(\hat{c}_k C + c_k A, \hat{c}_k C + c_k A) + \\ &6 \sum_{l=0}^{J-1} \sum_{k,j} \alpha_{i,J-l}^k \hat{a}_{k,j} C'(C'(\hat{c}_j C + c_j A)) + \\ &3 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k \hat{c}_k^2 C'(C^2) + 6 \sum_{l=0}^{J-1} \sum_k \alpha_{i,J-l}^k (\hat{c}_k c_k - \sum_j a_{k,j} \hat{c}_j) C'(CA) + \\ &6 \sum_{l=0}^{J-1} \sum_{k,j} \alpha_{i,J-l}^k a_{k,j} C'(A(\hat{c}_j C + c_j A)) - \sum_{l=0}^{J-1} \sum_{k,r} \alpha_{i,J-l}^k \alpha_{i,J-l}^r [C, C'(\hat{c}_r C + c_r A)]. \end{aligned} \quad (1.3.14)$$

Using (1.3.11) we obtain $\varphi_i^{(3)}(0)$ as reported in Table 1.1 and from (1.3.10) we obtain $Y_i^{(3)}(0)$ reported in Table 1.3.

By imposing $Y_{s+1}^{(3)}|_{h=0} = y^{(3)}|_{h=0}$ we obtain the conditions for order 3 (recalling that $\alpha_{s+1, J-l}^k = \beta^k$ and $\sum_{l=0}^{J-1} \beta_{J-l}^k = \hat{b}_k$) reported in Table 1.4.

1.4 Extra coupling conditions for order 4

For methods of order up to 3, we observe that the order conditions form a subset of the set of order conditions for the partitioned Runge-Kutta method and the commutator-free methods. The set of order conditions for methods of order 4 is very large, and we do not attempt to derive them here. Rather we investigate that there exist extra coupling conditions that do not belong to the set of order conditions for the partitioned Runge-Kutta method.

We consider coefficients of elementary differentials preceded by an A in both the expressions for the fourth derivatives ($q = 4$) of the exact and numerical solutions (1.2.1) and (1.2.2) respectively. That means matching the terms in $Ay^{(3)}|_{h=0}$ and $4 \sum_{i,j} a_{i,j} \varphi_{i,j}(0) AY_j^{(3)}(0)$, since $\varphi_{i,j}(0) = I$. We obtain

$$\begin{aligned} Ay^{(3)}|_{h=0} &= A \left[C''(C, C) + C''(C, A) + C''(A, C) + C''(A, A) + C'(C'(C)) \right. \\ &\quad + C'(C'(A)) + C'(C^2) + C'(CA) + C'(AC) + C'(A^2) + 2C'(C)C \\ &\quad + 2C'(C)A + 2C'(A)C + 2C'(A)A + CC'(C) + CC'(A) + AC'(C) \\ &\quad \left. + AC'(A) + (C + A)^3 \right] y_0. \end{aligned} \quad (1.4.1)$$

Substitute for $\varphi_{i,j}(0)$ and $Y_j^{(3)}(0)$ in $4 \sum_j \sum_j a_{i,j} \varphi_{i,j}^{(0)} AY_j^{(3)}(0)$, and select elementary differentials whose coefficients contain the CF coefficients $\alpha_{s+1, J-l}^k := \beta_{J-l}^k$. These include $ACC'(C)y_0$, $ACC'(A)y_0$, $AC'(C)Cy_0$, $AC'(A)Cy_0$, arising from the terms

$$\begin{aligned} &16 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j, J-r}^k \alpha_{j, J-l}^m A [C, C'(\hat{c}_m C + c_m A)] y_0 + \\ &8 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j, J-r}^k \alpha_{j, J-l}^m A [C, C'(\hat{c}_k C + c_k A)] y_0 \\ &- 4 \overline{\sum_{jlrk}} a_{i,j} \alpha_{j, J-l}^k \alpha_{j, J-l}^r A [C, C'(\hat{c}_r C + c_r A)] y_0 \\ &+ 16 \sum_{j,k} a_{i,j} \hat{c}_j \hat{a}_{j,k} AC'(\hat{c}_k C + c_k A) C y_0 + 8 \sum_{j,k} a_{i,j} \hat{c}_j \hat{a}_{j,k} ACC'(\hat{c}_k C + c_k A) y_0, \end{aligned} \quad (1.4.2)$$

where

$$\overline{\sum_{jlrk}} := \sum_j \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k. \quad (1.4.3)$$

Chapter 1. Order conditions for the semi-Lagrangian exponential integrators

Table 1.4: Conditions of order 3

	condition	elementary differential
commutators	$\left. \begin{aligned} 4 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \beta_{J-r}^k \sum_m \beta_{J-l}^m \hat{c}_m \\ 2 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \beta_{J-r}^k \sum_m \beta_{J-l}^m \hat{c}_k - \\ \sum_{l=0}^{J-1} \sum_k \beta_{J-l}^k \sum_r \beta_{J-l}^r \hat{c}_r = 0 \end{aligned} \right\}$	$[C'(C), C]$
	$\left. \begin{aligned} 4 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \beta_{J-r}^k \sum_m \beta_{J-l}^m c_m - \\ 2 \sum_{l=0}^{J-1} \sum_{r=0}^l \sum_k \beta_{J-r}^k \sum_m \beta_{J-l}^m c_k - \\ \sum_{l=0}^{J-1} \sum_k \beta_{J-l}^k \sum_r \beta_{J-l}^r c_r = 0 \end{aligned} \right\}$	$[C'(A), C]$
higher order differentials	$3 \sum_k \hat{b}_k \hat{c}_k^2 = 1$	$C''(C, C)$
	$3 \sum_k \hat{b}_k \hat{c}_k c_k = 1$	$C''(C, A)$
	$3 \sum_k \hat{b}_k \hat{c}_k c_k^2 = 1$	$C''(A, A)$
	$3 \sum_k \hat{b}_k c_k \hat{c}_k = 1$	$C''(A, C)$
	$6 \sum_k \hat{b}_k \sum_j \hat{a}_{k,j} \hat{c}_j = 1$	$C'(C'(C))$
	$6 \sum_k \hat{b}_k \sum_j \hat{a}_{k,j} c_j = 1$	$C'(C'(A))$
	$3 \sum_k \hat{b}_k \hat{c}_k^2 = 1$	$C'(C^2)$
	$6 \sum_k \hat{b}_k (\hat{c}_k c_k - \sum_j a_{k,j} \hat{c}_j) = 1$	$C'(CA)$
	$6 \sum_k \hat{b}_k \sum_j a_{k,j} \hat{c}_j = 1$	$C'(AC)$
	$6 \sum_k \hat{b}_k \sum_j a_{k,j} c_j = 1$	$C'(A^2)$
products of lower order differentials	$6 \sum_k \hat{b}_k (\sum_j b_j) \hat{c}_k - 6 \sum_j b_j \sum_k \hat{a}_{j,k} \hat{c}_k = 2$	$C'(C)A$
	$6 \sum_k \hat{b}_k (\sum_j b_j) c_k - 6 \sum_j b_j \sum_k \hat{a}_{j,k} c_k = 2$	$C'(A)A$
	$3 - 6 \sum_j b_j \hat{c}_j + 3 \sum_j b_j \hat{c}_j^2 = 1$	$C^2 A$
	$6 \sum_j b_j \hat{c}_j - 6 \sum_j b_j \hat{c}_j^2 = 1$	CAC
	$6 \sum_j b_j c_j - 6 \sum_j b_j \hat{c}_j c_j = 1$	CA^2
	$6 \sum_j b_j \sum_m \hat{a}_{j,m} \hat{c}_m = 1$	$AC'(C)$
	$6 \sum_j b_j \sum_m \hat{a}_{j,m} c_m = 1$	$AC'(A)$
	$3 \sum_j b_j \hat{c}_j^2 = 1$	AC^2
	$6 \sum_j b_j (\hat{c}_j c_j \sum_m a_{j,m} \hat{c}_m) = 1$	ACA
	$6 \sum_j b_j \sum_m a_{j,m} \hat{c}_m = 1$	$A^2 C$
	$6 \sum_j b_j \sum_m a_{j,m} c_m = 1$	A^3

Comparing coefficients of elementary differentials

$$ACC'(C)y_0, ACC'(A)y_0, AC'(C)Cy_0, AC'(A)Cy_0$$

we obtain the order conditions

$$\left. \begin{aligned} &16 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m \hat{c}_m - 8 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m \hat{c}_k \\ &- 4 \overline{\sum_{jlrk}} a_{i,j} \alpha_{j,J-l}^k \alpha_{j,J-l}^r \hat{c}_r + 8 \sum_{j,k} a_{i,j} \hat{c}_j \hat{a}_{j,k} \hat{c}_k = 1, \end{aligned} \right\} \quad (1.4.4)$$

$$\left. \begin{aligned} &16 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m c_m - 8 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m c_k \\ &- 4 \overline{\sum_{jlrk}} a_{i,j} \alpha_{j,J-l}^k \alpha_{j,J-l}^r c_r + 8 \sum_{j,k} a_{i,j} \hat{c}_j \hat{a}_{j,k} c_k = 1, \end{aligned} \right\} \quad (1.4.5)$$

$$\left. \begin{aligned} &- 16 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m \hat{c}_m + 8 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m \hat{c}_k \\ &+ 4 \overline{\sum_{jlrk}} a_{i,j} \alpha_{j,J-l}^k \alpha_{j,J-l}^r \hat{c}_r + 16 \sum_{j,k} a_{i,j} \hat{c}_j \hat{a}_{j,k} \hat{c}_k = 2, \end{aligned} \right\} \quad (1.4.6)$$

Simplifying (1.4.4)-(1.4.6) via the use of third order conditions we obtain the order conditions

$$4 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m \hat{c}_m - 2 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m \hat{c}_k - \overline{\sum_{jlrk}} a_{i,j} \alpha_{j,J-l}^k \alpha_{j,J-l}^r \hat{c}_r = 0, \quad (1.4.7)$$

$$4 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m c_m - 2 \overline{\sum_{jlrk}} \sum_m a_{i,j} \alpha_{j,J-r}^k \alpha_{j,J-l}^m c_k - \overline{\sum_{jlrk}} a_{i,j} \alpha_{j,J-l}^k \alpha_{j,J-l}^r c_r = 0, \quad (1.4.8)$$

where $i = s + 1$ so that $a_{i,j} = a_{s+1,j} = b_j$, $j = 1, \dots, s$.

Assuming that the RK tableaux (1.1.3) fulfill the order conditions for a classical partitioned RK method of order 4, and that the b 's are different from the \hat{b} 's, i.e., $b_j \neq \hat{b}_j$ for some $j = 1, \dots, s$, then the order conditions in (1.4.7)-(1.4.8) will result in a *new or extra set of coupling conditions* (involving the α and β coefficients) for our method, which are not included in the set of order conditions for the classical partitioned RK methods and CF methods. This is not the case for orders 1 through 3 where all our order conditions are only a subset of those for the partitioned RK and CF methods.

In figure 1.1(a) we have a numerical test showing the order in time for a fourth order method (DIRK-CF4). This method is constructed using the additive partitioned IMEX RK method of Kennedy and Carpenter [5] (here named as IMEX4) wherein we derive from

Chapter 1. Order conditions for the semi-Lagrangian exponential integrators

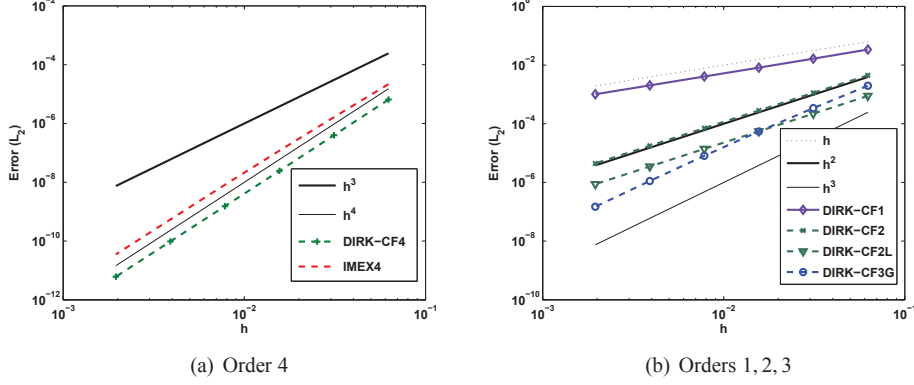


Figure 1.1: Numerical order tests using Burgers' equation, $u_t + uu_x = \nu u_{xx}$, on Dirichlet homogeneous BCs, $x \in [0, 1]$, $\nu = 0.05$, $u_0 = \sin \pi x$, $T = 1$, $\Delta x = 1/32$. Plot of the 2– norm error as a function of step size $h = 2^{-n}$, $n = 4, 5, \dots, 9$. **(a)** Order test for the fourth order DIRK-CF4 and IMEX4 **(b)** Order test for first order DIRK-CF1, second order DIRK-CF2 and DIRK-CF2L, third order DIRK-CF3G.

the corresponding explicit tableau the commutator-free coefficients via (1.1.2) satisfying the CF order conditions as given by Owren [6]. Using two exponentials in the last internal stage and two exponentials in the update stage, these conditions are given by

$$\frac{1}{6} \sum_k \beta_1^k c_k + \frac{1}{18} \sum_k \beta_2^k = \frac{1}{24} \quad (1.4.9)$$

$$\frac{1}{4} \sum_k \beta_1^k c_k^2 + \frac{1}{12} \sum_k \beta_2^k = \frac{1}{24} \quad (1.4.10)$$

$$\frac{1}{2} \sum_{j,k} \beta_1^k a_{k,j} c_j + \frac{1}{12} \sum_k \beta_2^k = \frac{1}{24} \quad (1.4.11)$$

$$\frac{1}{2} \sum_{j,k} b_j c_j \alpha_{j,1}^k c_k + \frac{1}{2} \sum_{i,j,k} b_i a_{i,j} c_j \alpha_{i,2}^k = \frac{1}{24}. \quad (1.4.12)$$

It is important to note, however, that our new method DIRK-CF4 automatically satisfies the coupling conditions (1.4.7)-(1.4.8) as part of the partitioned RK order conditions since in this choice of IMEX RK scheme the b 's and \hat{b} 's are the same (see [5]). The figure also shows a comparison between the DIRK-CF4 and its counterpart IMEX4. The numerical experiment is performed on the viscous Burgers' equation

$$u_t + uu_x = \nu u_{xx}$$

over a spatial domain $(0, 1)$ with initial condition $u(x, 0) = u_0(x) = \sin(\pi x)$, and homogeneous Dirichlet boundary conditions. We integrate on the interval $[0, T]$ ($T =$

1, $\nu = 0.05$ in this case) with time steps in the range $\{h = 2^{-n} | n = 4, 5, \dots, 9\}$. The spatial discretization is the standard centered differences on a uniform grid of mesh step $\Delta x = 1/32$. The error is measured as a grid-point error in the 2-norm, and the reference (exact) solution is computed as in [3]. Figure 1.1(b) shows the numerical order tests performed for some of the first to third order methods derived in [3].

References

- [1] E. Celledoni, *Eulerian and semi-Lagrangian commutator-free exponential integrators*, CRM Proceedings and Lecture Notes **39** (2005), 77–90.
- [2] E. Celledoni, *Semi-Lagrangian methods and new integrators for convection dominated problems*, Oberwolfach Reports **12** (2006).
- [3] E. Celledoni and B. K. Kometa, *Semi-Lagrangian Runge-Kutta exponential integrators for convection dominated problems*, J. Sci. Comput. **41** (2009), no. 1, 139–164.
- [4] E. Celledoni, A. Marthinsen, and B. Owren, *Commutator-free Lie group methods*, FGCS **19** (2003).
- [5] C. A. Kennedy and M. H. Carpenter, *Additive Runge-Kutta schemes for convection-diffusion-reaction equations*, Appl. Numer. Math. **44** (2003), no. 1-2, 139–181.
- [6] B. Owren, *Order conditions for commutator-free Lie group methods*, J. Phys. A **39** (2006), no. 19, 5585–5599. MR 2220777 (2007e:37054)

Paper III

Semi-Lagrangian multistep exponential integrators for index 2 differential-algebraic systems

Elena Celledoni and Bawfeh Kingsley Kometa

Published in *Journal of Computational Physics*, Vol. 230
(2011), pp. 3413–3429

Chapter 4

Semi-Lagrangian multistep exponential integrators for index 2 differential-algebraic systems

Abstract. Implicit-explicit (IMEX) multistep methods are very useful for the time discretization of convection diffusion PDE problems such as the Burgers' equations and the incompressible Navier-Stokes equations. In the latter as well as in PDE models of plasma physics and of electromechanical systems, semi-discretization in space gives rise to differential-algebraic (DAE) system of equations often of index higher than 1. In this paper we propose a new class of exponential integrators for index 2 DAEs arising from the semi-discretization of PDEs with a dominating and typically nonlinear convection term. This class of problems includes the incompressible Navier-Stokes equations. The integration methods are based on the backward differentiation formulae (BDF) and they can be applied without modifications in the semi-Lagrangian integration of convection diffusion problems. The approach gives improved performance at low viscosity regimes.

4.1 Introduction

We consider differential-algebraic equations (DAEs) of the form

$$\begin{aligned} \dot{y} &= C(y)y + f(y, z), \\ 0 &= g(y), \end{aligned} \tag{4.1.1}$$

with consistent initial data $y(t_0) = y_0$, $z(t_0) = z_0$, where $y = y(t) \in \mathbb{R}^n$, $z = z(t) \in \mathbb{R}^m$, for all $t \in [t_0, T]$; while $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$, $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$, and $C = C(y) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ is a matrix-valued function of y . The notation \dot{y} denotes the derivative with respect to

Chapter 4. Semi-Lagrangian multistep exponential integrators for index 2 DAEs

t . DAEs of this type arise, for example, from the semi-discretization (in space) of PDE models in plasma physics, and also of the incompressible Navier-Stokes equations. In this case $C(y)y$ represents the nonlinear convection term, $f(y, z)$ represents the diffusion and pressure terms and $g(y)$ comes from the incompressibility constraint; f and g are both linear in this case. Assuming (4.1.1) generally results from a convection diffusion PDE, we will refer to the term $C(y)y$ as the convecting vector field or simply the convection term.

The system of DAEs (4.1.1) is of *differential index 2* if the functions f, g are sufficiently differentiable and the matrix $g_y f_z$ is nonsingular in a neighbourhood of the solution. Here f_z and g_y denote Jacobian transformations of f and g . The algebraic part of (4.1.1) represents the main constraint. A second (hidden) constraint,

$$g_y(y)(C(y)y + f(y, z)) = 0, \quad (4.1.2)$$

is given by differentiating the main algebraic constraint with respect to t . The variable y is commonly referred to as the *differential or state* variable while the z -variable is the *algebraic or constraint* variable or simply the *Lagrange multiplier*. For more details about index 2 DAEs we refer to [17] for example.

Runge-Kutta (RK) methods have been considered for the time discretization of index 2 DAE systems (see [16, 2, 29, 17, 23, 24]). Some of these RK methods achieve high order of convergence with comparatively little storage requirements and have good stability properties. However, fully implicit RK methods generally have a drawback over the *IMEX*¹ or *DIRK*² methods in terms of computational costs per time step. For reasons of ease of implementation, we only wish to consider IMEX methods, whereby we treat the nonlinear term $C(y)y$ explicitly and the term $f(y, z)$ implicitly as it may be stiff and linear in some applications.

One-step IMEX methods for hyperbolic systems with relaxation have recently been studied in [33] and applied to index-1 DAEs in [10]. In the framework of IMEX one-step, exponential integrators, we have earlier considered *DIRK-CF* methods for convection diffusion PDE problems [7]. These methods are based on commutator-free³ (CF) Lie group methods and are typically constructed from IMEX RK methods where the implicit part is a DIRK method (see Appendix 4.5.3 for details on IMEX-CF exponential integrators and DIRK-CF methods in particular). The main reason for developing this type of methods is that they can be applied to the semi-Lagrangian numerical integration of convection diffusion PDEs, achieving improved stability behavior in the small viscosity limit.

When applied directly to problems of the type (4.1.1), the DIRK-CF methods can however only give convergence of order 2. The same remains true when using IMEX RK methods such as those in [3, 26]. DIRK methods for (4.1.1) may appear cheaper to

¹*Implicit-explicit* (IMEX): Methods that treat, for example, the term $C(y)y$ explicitly and the remaining terms implicitly.

²*Diagonally implicit RK* (DIRK): Implicit RK method with coefficients $\{a_{ij}\}$ such that $a_{ij} = 0$ for all $i < j$.

³using the terminology of [8]. See also Appendix 4.5.3 for further details.

implement than fully implicit RK methods, but the order of convergence is greatly limited by the stage order⁴. For example the DIRK methods with nonzero diagonal entries, e.g. most of the methods in [1, 3], will give convergence of order at most 2 (see [16, p.18] and [17, Lemma 4.4, Thm. 4.5, p.495-496]). All the DIRK methods in [26, 36, 27] have stage order at most 2, thus they would lead to convergence of order at most 3 (according to [22, Thm.5.2]).

On the other hand, linear k -step methods such as BDF methods, are known to give convergence of order $p = k$, for $1 \leq k \leq 6$, in both variables (see for example [17, VII.3]). They are also known to be A -stable for $1 \leq k \leq 2$ and $A(\alpha)$ -stable for $3 \leq k \leq 6$. We however do not wish to treat the whole system implicitly. IMEX multistep methods based on BDF schemes have been developed and applied for the time discretization of convection diffusion PDE problems such as the Burgers' equations (see for example [4]) as well as the incompressible Navier-Stokes equations (see [30, 25, 20, 34, 41, 14]).

We hereby propose a new class of exponential integrators for (4.1.1) based on the backward differentiation formula (BDF). We name these methods *BDF-CF* for short. They have about the same implementation ease as the DIRK-CF, and as such they can be regarded as their multistep counterpart. Their main advantage compared to the DIRK-CF is that they can achieve order of convergence higher than 2, when applied to (4.1.1) both in the algebraic and differential variables.

The methods are a subclass of the IMEX multistep methods and they are closely related to the SBDF methods presented and studied in [4, 21]. Compared to these methods the BDF-CF methods can be used without modifications in a semi-Lagrangian approach to convection diffusion problems, whereby the exponentials must be realized as flows of pure convection problems. The pure convection flows are typically implemented by tracing characteristics, as in [35, 41, 7], or by computing more accurate approximations of the pure convection flows as in [30, 40] (this can for example be achieved using an integration method of higher order).

We recall that the concept of multistep exponential integrators is not new to this paper. Explicit multistep exponential integrators have recently been studied for semilinear ODEs with a linear stiff term and a nonlinear non stiff term, see [11, 5, 32]. These methods require the computation of exponentials of the linear stiff term, and the obtained schemes are fully explicit. If the nonlinear term represents convection and is in the form $C(y)y$, the methods we present here can also be applied to such ODEs, but our assumption is that the problems are convection-dominated and therefore we treat the nonlinear term explicitly by exponentials (or accurate solution of pure convection flows) and the linear term implicitly. Commutator-free methods of multistep type were originally considered in [12].

⁴A RK method with coefficients $\{a_{ij}, b_i, c_i\}$, $i, j = 1, \dots, s$, has (internal) *stage order* q , if q is the greatest integer such that $\sum_{j=1}^s a_{ij}c_j^{k-1} = c_i^k/k$, $i = 1, \dots, s$ hold for all $k = 1, \dots, q$.

4.1.1 The BDF-CF methods

We define the k -step exponential BDF (or simply BDF-CF) method as follows: Given k initial values y_0, \dots, y_{k-1} , find (y_k, z_k) such that

$$\begin{aligned} \alpha_k y_k + \sum_{i=0}^{k-1} \alpha_i \varphi_i y_i &= hf(y_k, z_k), \\ 0 &= g(y_k) \end{aligned} \quad (4.1.3)$$

where $\varphi_i := \exp\left(\sum_{j=0}^{k-1} a_{i+1, j+1} hC(y_j)\right)$, $i = 0, \dots, k-1$, and $a_{ij} \in \mathbb{R}$, $i, j = 1, \dots, k$, are coefficients of the method, while α_i , $i = 0, \dots, k$, are coefficients of the linear k -step classical BDF method. The use of the functions φ_i (exact flows of the linearized convection term) is introduced to obtain improved performance in the treatment of convection-dominated problems. This idea was also found useful in the DIRK-CF methods [7] and in the multirate methods for atmospheric flow simulation [40]. We also refer to the methods as *commutator-free* (CF) multistep exponential integrators, since the flows of the convecting vector fields do not contain matrix commutators. Thus the name BDF-CF is used for the method (4.1.3). In a more general setting involving CF exponential integrators [6], the functions φ_i would be defined as a composition of convection flows. However, in the BDF-CF methods considered here single flows would suffice. More precisely we shall write a k -order (typically k -step) method as BDF k -CF. The overall method is termed *semi-Lagrangian* if we treat each flow, $\varphi_i y_i$, in a semi-Lagrangian fashion [35, 41] (see also [7, Sect.3.1]); and is found useful for the time integration of convection diffusion PDEs and the Navier-Stokes equations. Nevertheless, the flows can also be computed using other numerical methods such as the direct approximation of the matrix exponentials via a Padé approximant or by using a Krylov subspace method [9, 19, 38, 31, 28]. The semi-Lagrangian approach was shown [7] to be more stable and accurate than the latter two methods, in the solution of convection-dominated convection-diffusion problems. In this paper the semi-Lagrangian approach has been used in all numerical experiments involving time dependent PDEs.

Assuming once again that the system (4.1.1) arises from the semi-discretization (in space) of a PDE, then a close comparison of the BDF-CF methods with the operator-integrating-factor (OIF) splitting methods of Maday *et al.* [30] (also considered in [14]) will be as follows: The OIF states; find (y_k, z_k) such that

$$\alpha_k y_k + \sum_{i=0}^{k-1} \alpha_i \tilde{y}_i = hf(y_k, z_k), \quad g(y_k) = 0, \quad (4.1.4)$$

where α_i are coefficients of the classical k -step BDF method, and \tilde{y}_i are solutions of the linearized pure convection problems

$$\dot{\tilde{y}} = C(\tilde{y}_p)\tilde{y}, \quad t \in (t_i, t_k), \quad \tilde{y}(t_i) = y_i$$

where $\tilde{y}_p(t)$ is a $(k-1)$ -degree polynomial interpolation/extrapolation of the initial values. The BDF-CF methods are also stated as in (4.1.4), but the \tilde{y}_i are computed in a different way.

The rest of the paper is organized as follows. In Section 4.2 we present a derivation of the new class of methods. In Section 4.3 we state some convergence results for the methods and provide a numerical evidence for the convergence of methods up to order 4. We discuss the stability of the methods in Section 4.4, making comparisons with some well-known IMEX multistep methods in the literature (for example [4, 21]). Unless stated otherwise, we shall say that a method has ‘order’ p to refer to the temporal order of convergence of the method. Also we shall only consider constant time steps, which shall be written as $h := \Delta t$. Given initial time t_0 , we shall write t_n to denote time level n such that $t_n := t_0 + nh$. For a given field variable $v = v(t)$ we denote the numerical approximation at time t_n by $v_n \approx v(t_n)$. In general we shall use the notation $\|\cdot\|$ for an arbitrary but well-defined norm of a vector or function.

4.2 Construction of the methods

Given a discrete time interval t_0, \dots, t_K (with final time $T = t_K$) and initial data y_0, \dots, y_{k-1} , $1 \leq k \leq K$, we describe a k -step BDF-CF method as follows

Algorithm 4.1. *BDF-CF method*

```

for  $n = k - 1$  to  $K - 1$  do
     $\varphi_i = \exp\left(h \sum_{j=1}^k a_{i+1,j} C(y_{n-k+j})\right)$ ,  $i = 0, \dots, k - 1$ ,
     $\alpha_k y_{n+1} + \sum_{i=0}^{k-1} \alpha_i \varphi_i y_{n+1-k+i} = hf(y_{n+1}, z_{n+1})$ ,
     $0 = g(y_{n+1})$ 
end

```

where $a_{i,j} \in \mathbb{R}$, $i, j = 1, \dots, k$, are coefficients of the BDF-CF method and α_i are coefficients of the classical k -step BDF method. Thus one can represent a k -step BDF-CF method in terms of its coefficients as in the following table

$$\begin{array}{c|ccc}
 y_{n-k+1} & a_{1,1} & \dots & a_{1,k} \\
 \vdots & \vdots & \dots & \vdots \\
 y_n & a_{k,1} & \dots & a_{k,k} \\
 \hline
 & C(y_{n-k+1}) & \dots & C(y_n)
 \end{array}$$

So that for each $n \geq k - 1$ the method solves for the unknown values, y_{n+1}, z_{n+1} , given the initial values y_{n-k+1}, \dots, y_n . For reasons of convenience (but without loss of generality)

Chapter 4. Semi-Lagrangian multistep exponential integrators for index 2 DAEs

we shall often drop the index n or simply treat the case with $n = k - 1$ as in (4.1.3). The first order (one-step) BDF-CF method is simply the semi-explicit backward Euler method, obtained by choosing $\varphi_0 = \exp(hC(y_n))$ in Algorithm 4.1. We shall therefore only consider k -step methods, for $k \geq 2$.

For simplicity we shall restrict the analysis of the methods to an ODE of the form

$$\dot{y} = C(y)y + f(y). \quad (4.2.1)$$

Extension to the DAE (4.1.1) is more or less direct.

Let us denote the exact value at time t_j by $\hat{y}_j := y(t_j)$, $j = 0, \dots, k$, and write $\hat{\varphi}_i := \exp\left(h \sum_{j=0}^{k-1} a_{i+1,j+1} C(y(t_j))\right)$, $i = 0, \dots, k-1$. Also let $\dot{\hat{y}}_j, \ddot{\hat{y}}_j, \dots$ denote the derivatives with respect to the time variable.

4.2.1 Second order method (BDF2-CF)

The *truncation error* $\tau_2(h)$ for a two-step method is given by

$$\frac{1}{h} \left[\frac{3}{2} \hat{y}_2 - 2\hat{\varphi}_1 \hat{y}_1 + \frac{1}{2} \hat{\varphi}_0 \hat{y}_0 \right] = f(\hat{y}_2) + \tau_2(h). \quad (4.2.2)$$

For a classical second order BDF method we have

$$\frac{1}{h} \left[\frac{3}{2} \hat{y}_2 - 2\hat{y}_1 + \frac{1}{2} \hat{y}_0 \right] = C(\hat{y}_2) \hat{y}_2 + f(\hat{y}_2) + \mathcal{O}(h^2). \quad (4.2.3)$$

Eliminate $f(\hat{y}_2)$ from (4.2.2) and (4.2.3) and put $\tau_2(h) = \mathcal{O}(h^2)$ to obtain

$$\frac{1}{h} \left[2\hat{\varphi}_1 \hat{y}_1 - \frac{1}{2} \hat{\varphi}_0 \hat{y}_0 - 2\hat{y}_1 + \frac{1}{2} \hat{y}_0 \right] - C(\hat{y}_2) \hat{y}_2 = \mathcal{O}(h^2), \quad (4.2.4)$$

which is a reasonable requirement for a second order method.

Putting

$$\begin{aligned} \hat{y}_0 &= \hat{y}_1 - h\dot{\hat{y}}_1 + \mathcal{O}(h^2), \\ \hat{y}_2 &= \hat{y}_1 + h\dot{\hat{y}}_1 + \mathcal{O}(h^2), \end{aligned}$$

we get via Taylor expansion (about $t = t_1$)

$$\begin{aligned} C(\hat{y}_2) \hat{y}_2 &= C(\hat{y}_1) \hat{y}_1 + hC(\hat{y}_1) \dot{\hat{y}}_1 + hC'(\hat{y}_1) (\dot{\hat{y}}_1) \hat{y}_1 + \mathcal{O}(h^2), \\ \hat{\varphi}_0 \hat{y}_0 &= \hat{y}_0 + a_{11} h [C(\hat{y}_1) - hC'(\hat{y}_1) (\dot{\hat{y}}_1)] (\hat{y}_1 + h\dot{\hat{y}}_1) + a_{12} h C(\hat{y}_1) (\hat{y}_1 + h\dot{\hat{y}}_1) \\ &\quad + \frac{h^2}{2} (a_{11} + a_{12})^2 C^2(\hat{y}_1) \hat{y}_1 + \mathcal{O}(h^3), \\ \hat{\varphi}_1 \hat{y}_1 &= \hat{y}_1 + a_{21} h [C(\hat{y}_1) - hC'(\hat{y}_1) (\dot{\hat{y}}_1)] \hat{y}_1 + a_{22} h C(\hat{y}_1) \hat{y}_1 \\ &\quad + \frac{h^2}{2} (a_{21} + a_{22})^2 C^2(\hat{y}_1) \hat{y}_1 + \mathcal{O}(h^3). \end{aligned}$$

Substituting into (4.2.4) and comparing coefficients of like terms and powers of h we obtain the following order conditions on the coefficients for order 2

$$\begin{aligned}
 2(a_{21} + a_{22}) - \frac{1}{2}(a_{11} + a_{12}) - 1 &= 0, \\
 -2a_{21} + \frac{1}{2}a_{11} - 1 &= 0, \\
 \frac{1}{2}(a_{11} + a_{12}) - 1 &= 0, \\
 (a_{21} + a_{22})^2 - \frac{1}{4}(a_{11} + a_{12})^2 &= 0.
 \end{aligned} \tag{4.2.5}$$

Solving this system yields a one-parameter set of coefficients, illustrated in the following table

y_{n-1}	$2(1 + 2\gamma)$	-4γ
y_n	γ	$1 - \gamma$
	$C(y_{n-1})$	$C(y_n)$

from which we define the second order BDF2-CF methods as

$$\frac{3}{2}y_{n+1} - 2\varphi_1 y_n + \frac{1}{2}\varphi_0 y_{n-1} = hf(y_{n+1}), \quad n \geq 1, \tag{4.2.6}$$

where $\varphi_0 = \exp(2(1 + \gamma)hC(y_{n-1}) - 4\gamma hC(y_n))$ and $\varphi_1 = \exp(\gamma hC(y_{n-1}) + (1 - \gamma)hC(y_n))$. Applied to the DAE (4.1.1) we get

$$\begin{aligned}
 \frac{1}{h} \left[\frac{3}{2}y_{n+1} - 2\varphi_1 y_n + \frac{1}{2}\varphi_0 y_{n-1} \right] &= f(y_{n+1}, z_{n+1}), \\
 0 &= g(y_{n+1}).
 \end{aligned} \tag{4.2.7}$$

Note that the value of the parameter γ can be arbitrarily chosen. However an optimal choice is possible in the ODE case. For example, in BDF2-CF, the local truncation error $\tau_2(h)$ can be minimized. We have from (4.2.2), using Taylor expansion about $t = t_1$, that

$$\begin{aligned}
 \tau_2(h) &= \frac{h^2}{8}\ddot{y}_1 - h^2 a_{21} \left[C'(\dot{y}_1) + C''(\dot{y}_1, \dot{y}_1) - CC'(\dot{y}_1) \right] \dot{y}_1 \\
 &= \frac{h^2}{2} \left[a_{11}(C'(\dot{y}_1) + C''(\dot{y}_1, \dot{y}_1))\dot{y}_1 - 4C^2\dot{y}_1 - 2a_{11}CC'(\dot{y}_1)\dot{y}_1 \right] \\
 &= h^2 \left[a_{11}C'(\dot{y}_1)\dot{y}_1 + C\ddot{y}_1 \right] - \frac{h^2}{2} \left(f'(\dot{y}_1) + f''(\dot{y}_1, \dot{y}_1) \right),
 \end{aligned}$$

Chapter 4. Semi-Lagrangian multistep exponential integrators for index 2 DAEs

which implies that for some constant c

$$\|\tau_2(h)\| \leq ch^2(1 + |a_{11}| + |a_{21}|) = ch^2(1 + |\gamma| + 2|1 + 2\gamma|).$$

Thus, the parameter γ can then be chosen so as to minimize $(1 + |\gamma| + 2|1 + 2\gamma|)$. This minimum is $\frac{3}{2}$ and occurs at $\gamma = -\frac{1}{2}$

4.2.2 Third order method (BDF3-CF)

The truncation error $\tau_3(h)$ for a three-step method is given by

$$\frac{1}{h} \left[\frac{11}{6}\hat{y}_3 - 3\hat{\varphi}_2\hat{y}_2 + \frac{3}{2}\hat{\varphi}_1\hat{y}_1 - \frac{1}{3}\hat{\varphi}_0\hat{y}_0 \right] = f(\hat{y}_3) + \tau_3(h). \quad (4.2.8)$$

A classical third order BDF method will satisfy

$$\frac{1}{h} \left[\frac{11}{6}\hat{y}_3 - 3\hat{y}_2 + \frac{3}{2}\hat{y}_1 - \frac{1}{3}\hat{y}_0 \right] = C(\hat{y}_3)\hat{y}_3 + f(\hat{y}_3) + \mathcal{O}(h^3). \quad (4.2.9)$$

Combining (4.2.8) and (4.2.9), and requiring that $\tau_3(h) = \mathcal{O}(h^3)$ we get

$$\frac{1}{h} \left[3\hat{\varphi}_2\hat{y}_2 - \frac{3}{2}\hat{\varphi}_1\hat{y}_1 + \frac{1}{3}\hat{\varphi}_0\hat{y}_0 - 3\hat{y}_2 + \frac{3}{2}\hat{y}_1 - \frac{1}{3}\hat{y}_0 \right] - C(\hat{y}_3)\hat{y}_3 = \mathcal{O}(h^3). \quad (4.2.10)$$

We put in (4.2.10)

$$\hat{y}_0 = \hat{y}_1 - h\dot{\hat{y}}_1 + \frac{h^2}{2}\ddot{\hat{y}}_1 + \mathcal{O}(h^3),$$

$$\hat{y}_2 = \hat{y}_1 + h\dot{\hat{y}}_1 + \frac{h^2}{2}\ddot{\hat{y}}_1 + \mathcal{O}(h^3),$$

$$\hat{y}_3 = \hat{y}_1 + 2h\dot{\hat{y}}_1 + 2h^2\ddot{\hat{y}}_1 + \mathcal{O}(h^3),$$

and carry out a Taylor expansion (about $t = t_1$). Comparing coefficients of like terms and powers of h we obtain the order conditions for order 3, comprising of 10 linearly dependent equations in 9 unknowns (see Appendix 4.5.2). Solving the system of equations in Maple yields a three-parameter family of methods, illustrated in the following table

y_{n-2}	$\frac{33}{2} - \frac{9}{4}\beta - 9\gamma$	$-18 + 9\alpha + \frac{9}{2}\beta + 9\gamma$	$\frac{9}{2} - 9\alpha - \frac{9}{4}\beta$
y_{n-1}	$3 + 2\alpha - \frac{1}{2}\beta - 2\gamma$	β	$-1 - 2\alpha - \frac{1}{2}\beta + 2\gamma$
y_n	α	$1 - \alpha - \gamma$	γ
	$C(y_{n-2})$	$C(y_{n-1})$	$C(y_n)$

from which the third order BDF3-CF methods are defined for $n \geq 2$.

4.2.3 Fourth order method (BDF4-CF)

We determine the coefficients, $\{a_{ij}\}$, $i, j = 1, \dots, 4$, for the fourth order method by requiring that the equation

$$\frac{1}{h} \left[4\hat{\varphi}_3\hat{y}_3 - 3\hat{\varphi}_2\hat{y}_2 + \frac{4}{3}\hat{\varphi}_1\hat{y}_1 - \frac{1}{4}\hat{\varphi}_0\hat{y}_0 - 4\hat{y}_3 + 3\hat{y}_2 - \frac{4}{3}\hat{y}_1 + \frac{1}{4}\hat{y}_0 \right] - C(\hat{y}_4)\hat{y}_4 = \mathcal{O}(h^4) \quad (4.2.11)$$

is satisfied. Again using Taylor expansion and comparing coefficients of like terms we obtain a 6-parameter set of coefficients given by

y_{n-3}	$4\alpha - 4\sigma - 8\varrho + 12 + \gamma + 2\kappa$	$-4\alpha + 8\varrho - 2\gamma - 3\kappa - 8 + 4\sigma$
y_{n-2}	$-3\beta + 3\alpha - \frac{3}{2}\varrho + \frac{3}{16}\gamma + \frac{3}{8}\kappa - \frac{3}{4}\sigma + \frac{3}{2}$	$9\beta - \frac{9}{2}\alpha - \frac{9}{8}\varrho - \frac{9}{32}\gamma - \frac{9}{32}\kappa - \frac{9}{8}\sigma + \frac{21}{4}$
y_{n-1}	α	$2 - \varrho - \sigma - \alpha$
y_n	β	$\frac{1}{4} - 3\beta + \frac{1}{2}\alpha + \frac{1}{8}\varrho - \frac{3}{32}\kappa - \frac{1}{32}\gamma - \frac{1}{8}\sigma$
	$C(y_{n-3})$	$C(y_{n-2})$
	γ	κ
	$-9\beta + \frac{9}{4}\alpha + \frac{9}{4}\varrho - \frac{9}{16}\kappa + \frac{9}{4}\sigma - \frac{9}{2}$	$\frac{3}{8}\varrho + 3\beta - \frac{3}{4}\alpha + \frac{3}{32}\gamma + \frac{15}{32}\kappa - \frac{3}{8}\sigma + \frac{3}{4}$
	σ	ϱ
...	$3\beta - \frac{3}{4}\alpha - \frac{3}{4}\varrho + \frac{1}{16}\gamma + \frac{3}{16}\kappa$	$-\beta + \frac{1}{4}\alpha + \frac{5}{8}\varrho - \frac{1}{32}\gamma - \frac{3}{32}\kappa + \frac{1}{8}\sigma + \frac{3}{4}$
	$C(y_{n-1})$	$C(y_n)$

defining the fourth order method for $n \geq 3$.

A similar procedure can be used to design BDF-CF methods of order up to 6. It is not yet clear if one can obtain stable methods of order higher than 6 in this manner, since the classical k -step BDF methods are only stable up to $k \leq 6$.

4.3 Some Convergence Results for the BDF-CF methods

We shall follow the strategy used by Hairer *et al.* [16, 17] to justify the convergence of the BDF-CF methods (4.1.3) when applied to the DAE (4.1.1). We begin with an existence and uniqueness theorem similar to the one in [17, Thm3.1,p.482].

Theorem 4.3.1. *Suppose that the initial values $y_j, z_j, j = 0, \dots, k-1$, satisfy*

$$y_j - y(t_j) = \mathcal{O}(h), \quad z_j - z(t_j) = \mathcal{O}(h), \quad g(y_j) = \mathcal{O}(h^2). \quad (4.3.1)$$

Chapter 4. Semi-Lagrangian multistep exponential integrators for index 2 DAEs

Then the nonlinear system

$$\begin{aligned} \alpha_k y_k + \sum_{i=0}^{k-1} \alpha_i \varphi_i y_i &= hf(y_k, z_k), \\ \mathbf{0} &= g(y_k) \end{aligned} \quad (4.3.2)$$

as in (4.1.3) with $\alpha_k \neq 0$, has a solution for $h \leq h_0$. Furthermore, this solution is unique and satisfies

$$y_k - y(t_k) = \mathcal{O}(h), \quad z_k - z(t_k) = \mathcal{O}(h). \quad (4.3.3)$$

The proof follows the pattern used by Hairer and Wanner [17, Thm3.1, p.482] with minor modifications.

Proof. We set

$$\eta = - \sum_{i=0}^{k-1} \frac{\alpha_i}{\alpha_k} \varphi_i y_i, \quad (4.3.4)$$

and define ζ close to $z(t_k)$ such that

$$g_y(\eta)(f(\eta, \zeta) + C(\eta)\eta) = \mathcal{O}(h). \quad (4.3.5)$$

We then replace h/α_k by a new step size which we again denote by h , without loss of generality. The system (4.3.2) becomes equivalent to

$$\begin{aligned} y_k &= \eta + hf(y_k, z_k), \\ \mathbf{0} &= g(y_k), \end{aligned} \quad (4.3.6)$$

which is the backward Euler method with initial data (η, ζ) . Thus we can apply ‘‘Theorem 3.1’’ of [16, p.31] or [17, Thm.3.1, p.482] to conclude the proof. Therefore it only suffices to show that

$$\eta - y(t_k) = \mathcal{O}(h), \quad \zeta - z(t_k) = \mathcal{O}(h), \quad g(\eta) = \mathcal{O}(h^2). \quad (4.3.7)$$

- (a) The first part of (4.3.7) follows by using that $\varphi_i y_i = y_i + \mathcal{O}(h)$ and $\sum_{i=0}^k \alpha_i = 0$, together with the assumptions in (4.3.1). Thus we get

$$\begin{aligned} \eta - y(t_k) &= -\frac{1}{\alpha_k} \sum_{i=0}^{k-1} \alpha_i (\varphi_i y_i - y(t_k)) \\ &= -\frac{1}{\alpha_k} \sum_{i=0}^{k-1} \alpha_i (y_i - y(t_k)) + \mathcal{O}(h) \\ &= -\frac{1}{\alpha_k} \sum_{i=0}^{k-1} \alpha_i [(y_i - y(t_i)) + (y(t_i) - y(t_k))] + \mathcal{O}(h). \end{aligned}$$

So that

$$\eta - y(t_k) = \mathcal{O}(h). \quad (4.3.8)$$

(b) Lastly, using the constraint (4.1.2) and the fact that $g_y f_z$ is invertible, we see (via Taylor expansion) that

$$g_y(\eta)(f(\eta, \zeta) + C(\eta)\eta) = g_y(y(t_k))f_z(y(t_k), z(t_k)) \cdot (\zeta - z(t_k)) + \mathcal{O}(\|\eta - y(t_k)\|). \quad (4.3.9)$$

Inserting (4.3.5) and (4.3.8) we get

$$\zeta - z(t_k) = \mathcal{O}(h).$$

(c) The proof of the third part of (4.3.7) follows exactly as in [17, Thm3.1,p.482].

□

The next theorem, which is proved exactly as in [17, Thm3.2, p.484], considers the influence of perturbations in the application of BDF-CF methods to (4.1.1).

Theorem 4.3.2. *Suppose y_k, z_k are given by (4.3.2) and consider perturbed values \hat{y}_k, \hat{z}_k satisfying*

$$\begin{aligned} \alpha_k \hat{y}_k + \sum_{i=0}^{k-1} \alpha_i \hat{\varphi}_i \hat{y}_i &= hf(\hat{y}_k, \hat{z}_k) + h\delta, \\ 0 &= g(\hat{y}_k) + \theta \end{aligned} \quad (4.3.10)$$

where $\hat{\varphi}_i := \exp\left(\sum_{j=0}^{k-1} a_{i+1,j+1} hC(\hat{y}_j)\right)$, $i = 0, \dots, k-1$. In addition to the assumptions of Theorem 4.3.1, suppose that for $j = 0, \dots, k-1$,

$$\hat{y}_j - y_j = \mathcal{O}(h^2), \quad \hat{z}_j - z_j = \mathcal{O}(h), \quad \delta = \mathcal{O}(h), \quad \theta = \mathcal{O}(h^2). \quad (4.3.11)$$

Then, for $h \leq h_0$, we have the estimates

$$\begin{cases} \|\hat{y}_k - y_k\| \leq \text{Const} (\|\Psi\Delta Y_0\| + h\|\Delta Z_0\| + h\|\delta\| + \|\theta\|), \\ \|\hat{z}_k - z_k\| \leq \frac{\text{Const}}{h} \left(\sum_{j=0}^{k-1} \|g_y(\hat{y}_k)(\hat{\varphi}_j \hat{y}_j - \varphi_j y_j)\| + h\|\Psi\Delta Y_0\| + h\|\Delta Z_0\| + h\|\delta\| + \|\theta\| \right) \end{cases} \quad (4.3.12)$$

where

$$\begin{aligned} \Delta Y_0 &:= (\hat{y}_{k-1} - y_{k-1}, \dots, \hat{y}_0 - y_0)^T, \quad \Delta Z_0 := (\hat{z}_{k-1} - z_{k-1}, \dots, \hat{z}_0 - z_0)^T \\ \Psi\Delta Y_0 &:= (\hat{\varphi}_{k-1} \hat{y}_{k-1} - \varphi_{k-1} y_{k-1}, \dots, \hat{\varphi}_0 \hat{y}_0 - \varphi_0 y_0)^T, \end{aligned}$$

while

$$\|\Psi\Delta Y_0\| := \max_{0 \leq j \leq k-1} \|\hat{\varphi}_j \hat{y}_j - \varphi_j y_j\|, \quad \|\Delta Z_0\| := \max_{0 \leq j \leq k-1} \|\hat{z}_j - z_j\|.$$

4.3.1 Local error

Suppose we consider exact initial values $y_j = y(t_j)$, $z_j = z(t_j)$, $j = 0, \dots, k-1$, in the BDF-CF formula (4.3.2) and also choose in (4.3.10) $\hat{y}_j = y(t_j)$, $\hat{z}_j = z(t_j)$, $j = 0, \dots, k$. Then we will have from (4.3.10) that $\theta = 0$, and by the construction of the BDF-CF methods the truncation error gives $\delta = \mathcal{O}(h^p)$. Also, since we now have $y_j = \hat{y}_j$, $z_j = \hat{z}_j$ for $j < k$, we get the following local error estimate, as a consequence of the estimates of Theorem 4.3.2.

Theorem 4.3.3. *Suppose that the BDF-CF method (4.3.2) applied to the DAE (4.1.1) has a truncation error of order p (in the sense implied by (4.2.2)). Then its local error satisfies*

$$y_k - y(t_k) = \mathcal{O}(h^{p+1}), \quad z_k - z(t_k) = \mathcal{O}(h^p). \quad (4.3.13)$$

4.3.2 Global Error

We observe that the convergence of the BDF-CF methods will require that the matrix-valued function $C(y)$ is sufficiently smooth on the space spanned by the initial data at each advancement in time.

Remark 4.3.4. We have the following remarks on the global convergence of the methods.

- (a) The result in Theorem 4.3.3 is still obtainable if we replace the terms $\|\Psi(\hat{Y}_0 - Y_0)\|$ and $\|g_y(\hat{y}_k)(\hat{\varphi}_j \hat{y}_j - \varphi_j y_j)\|$, $j = 0, \dots, k-1$, in (4.3.12) by the approximation (linearization)

$$\begin{aligned} \|g_y(\hat{y}_k)(\hat{\varphi}_j \hat{y}_j - \varphi_j y_j)\| &\leq \|g_y(\hat{y}_k)(\hat{y}_j - y_j)\| + \mathcal{O}(h \|g_y(\hat{y}_k) \hat{y}_j - y_j\|), \\ \|\Psi \Delta Y_0\| &\leq \|\Delta Y_0\| + \mathcal{O}(h \|\Delta Y_0\|) \end{aligned} \quad (4.3.14)$$

where $\|\Delta Y_0\| := \max_{0 \leq j \leq k-1} \|\hat{y}_j - y_j\|$. Such approximations are possible by using Taylor expansion methods, which in turn depend on the smoothness of the function $C(y)$.

- (b) Using (4.3.14) appropriately we can follow the same proof as ‘‘Theorem 3.5’’ of [17, p.486] to obtain the convergence of the BDF-CF applied to the index 2 DAE (4.1.1). Thus according to ‘‘Theorem 3.5’’ of [17, p.486] we expect to get convergence of order $p = k$, for $k \leq 6$, in both the algebraic and differential variables, upon applying the k -step BDF-CF method as detailed out in Algorithm 4.1 on page 73. This is investigated numerically in the following two subsections.

4.3.3 Numerical example

We consider the index 2 problem (see [18])

$$\begin{cases} \dot{y}_1 = y_1^2 + z + \cos t - 1, \\ \dot{y}_2 = y_1^2 + y_2^2 - \sin t - 1, \\ 0 = y_1^2 + y_2^2 - 1, \end{cases} \quad t \in [1, 2], \quad (4.3.15)$$

with the exact solution given by

$$y_1(t) = \sin t, \quad y_2(t) = \cos t, \quad z(t) = \cos^2 t. \quad (4.3.16)$$

This DAE is comparable to (4.1.1) with $y = (y_1, y_2)^T$, $g(y) = y_1^2 + y_2^2 - 1$,

$$C(y) = \begin{pmatrix} y_1 & 0 \\ y_1 & y_2 \end{pmatrix}, \quad f = f(t, y, z) = \begin{pmatrix} z + \cos t - 1 \\ -\sin t - 1 \end{pmatrix}.$$

We now solve (4.3.15) using each of the methods BDF1-CF, BDF2-CF ($\gamma = 0$), BDF3-CF ($\alpha = \beta = \gamma = 0$) and BDF4-CF ($\alpha = \beta = \gamma = \sigma = \varrho = \kappa = 0$). Since the DAE system is small, we have computed the matrix exponentials using MATLAB's built in `expm` function. We have used the exact solution (4.3.16) to find the starting values for the multistep methods. The global error (in the discrete L_2 -norm, see Appendix 4.5.1) at time $T = 2$, is plotted as a function of time step h , taking $h = 1/2^r$, $r = 4, \dots, 11$. As shown in Figure 4.1, we observe that for $k = 1, \dots, 4$, the method BDF k -CF gives convergence of order $p = k$ in both the differential and algebraic variables y and z . This agrees with the conclusion in Remark 4.3.4 for $k \leq 4$.

4.3.4 Numerical test on Navier-Stokes

Next we consider the incompressible Navier-Stokes equations in \mathbb{R}^2 ,

$$\begin{aligned} \mathbf{u}_t + (\mathbf{u} \cdot \nabla) \mathbf{u} &= -\nabla \bar{p} + \frac{1}{Re} \nabla^2 \mathbf{u}, & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} &= 0, & \text{in } \Omega, \end{aligned} \quad (4.3.17)$$

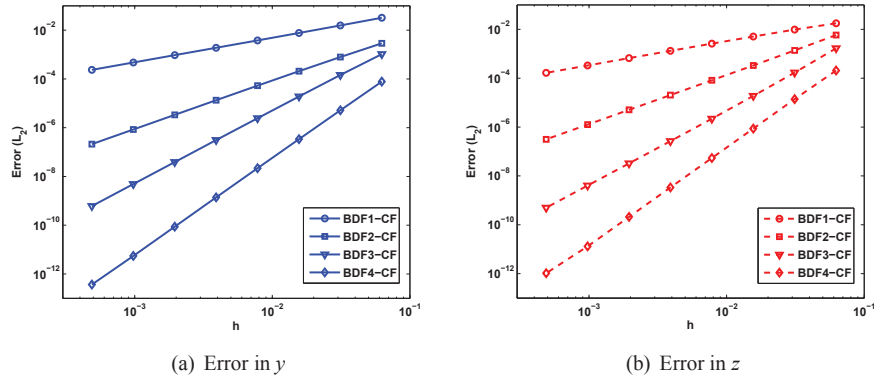


Figure 4.1: Order of different BDF-CF methods to index 2 DAE (4.3.15). Errors are measured (in the discrete L_2 -norm) at time $T = 2$ as functions of time step $h = 1/2^r$, $r = 4, \dots, 11$. Methods of order 1, 2, 3, 4 are represented by circles (\circ), squares (\square), triangles (∇) and diamonds (\diamond) respectively. **(a)** shows the errors in the differential variable y , while **(b)** shows the errors in the algebraic variable z .

Chapter 4. Semi-Lagrangian multistep exponential integrators for index 2 DAEs

with prescribed initial data and velocity boundary conditions. The constant Re is the Reynolds number, $\mathbf{x} = (x_1, x_2)^T \in \Omega \subset \mathbb{R}^2$, $t \in [0, T]$, while $\mathbf{u} = \mathbf{u}(\mathbf{x}, t) = (u_1, u_2)^T \in \mathbb{R}^2$ is the fluid velocity and $\bar{p} = \bar{p}(\mathbf{x}, t) \in \mathbb{R}$ is the pressure.

For the spatial discretization we employ a spectral element method (SEM) based on a standard Galerkin weak formulation and we use rectangular elements. The approximation is done in $\mathbb{P}_N - \mathbb{P}_{N-2}$ compatible velocity-pressure discrete spaces, i.e., keeping the time variable t fixed, we approximate the velocity by a N -degree Lagrange polynomial based on Gauss-Lobatto-Legendre (GLL) nodes in each spatial coordinate, and the pressure by $(N - 2)$ -degree Lagrange polynomial based on Gauss-Legendre (GL) nodes. The discrete spaces are spanned by tensor product polynomial basis functions. A more vivid description of this type of spatial discretization of Navier-Stokes is given by Fischer *et al.* [14]. The result is a semi-discrete (time-dependent) system of equations

$$\begin{aligned} B\dot{y} + C(y)y + Ay - D^T z &= 0, \\ Dy &= 0 \end{aligned} \tag{4.3.18}$$

where $y = y(t) \in \mathbb{R}^n$, $z = z(t) \in \mathbb{R}^m$, represent the discrete velocity and pressure respectively, while the matrices A, B, C, D, D^T represent the discrete Poisson (negative Laplace), mass, convection, divergence and gradient operators respectively. Meanwhile the degrees of freedom m and n depend on the polynomial degree N , the dimension of the computational domain (2 in our case) and the imposed boundary conditions. A detailed description of these discrete operators can be found in [13, 14]. The system (4.3.18) satisfies the requirements of the index 2 DAE (4.1.1), with $f(y, z) = B^{-1}(Ay - D^T z)$, $g(y) = Dy$, linear in their arguments. The matrix $g_y f_z = DB^{-1}D^T$ is invertible since B is positive definite. In fact, given $w \in \mathbb{R}^m$, $w \neq 0$, we have that

$$w^T (g_y f_z) w = (D^T w)^T B^{-1} D^T w > 0,$$

making $DB^{-1}D^T$ positive definite (assuming that the compatibility of the discrete spaces makes D^T to be of full rank). Thus the BDF-CF methods are applicable for the time integration of (4.3.18). Application of a BDF-CF method will result in a discrete Stokes system of equations that can be resolved via a block-LU decomposition (see [13]).

As a test example we consider the Taylor vortex problem [30, 37], with exact (analytic) solution given by

$$\left. \begin{aligned} u_1 &= -\cos(\pi x_1) \sin(\pi x_2) \exp(-2\pi^2 t/Re), \\ u_2 &= \sin(\pi x_1) \cos(\pi x_2) \exp(-2\pi^2 t/Re), \\ \bar{p} &= -\frac{1}{4} [\cos(2\pi x_1) + \cos(2\pi x_2)] \exp(-4\pi^2 t/Re). \end{aligned} \right\} \tag{4.3.19}$$

In this example we have used Dirichlet boundary conditions on the spatial domain $\Omega = [-1, 1]^2$, spectral element discretization (SEM) of order $N = 12$ with $Ne = 4$ rectangular elements, and the time integration is done up to time $T = 1$ using different constant step

sizes $h = T/2^r$, $r = 4, \dots, 9$. In our implementation of the semi-Lagrangian method the work of tracking characteristics and making interpolations is done element-by-element as in the SLSE methods of [41] (See also [15]). We have used the exact solution (4.3.19) to find the starting values for the multistep methods. The error in both time and space is measured. The error (at time T) in the velocity is measured in the H_1 -norm and the error in the pressure is measured in the L_2 -norm (see Appendix 4.5.1 for description of these norms). Figure 4.2 shows the temporal orders of convergence obtained with the methods BDF1-CF, BDF2-CF (with $\gamma = 1/3$) and BDF3-CF (with $\alpha = \beta = \gamma = 0$) applied to the semi-discrete incompressible Navier-Stokes problem (4.3.18). The Reynolds number used is $Re = 2\pi^2$. The same example was also used to test the fourth order method, BDF4-CF (not included in the figures), which showed a better overall convergence than the lower order methods. In this case, however, the temporal error is no longer dominant over the spatial error, and the overall error (both in time and space) is no longer monotonic with respect to h . Figure 4.3 shows the spectral convergence in space.

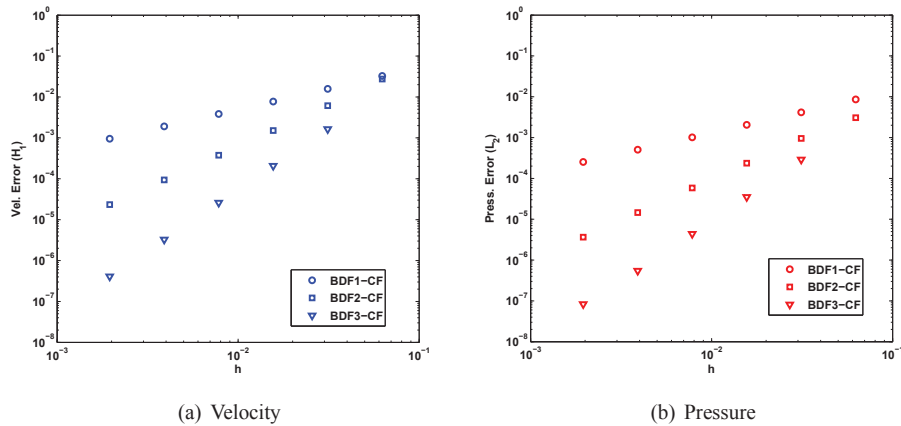


Figure 4.2: Temporal order test of different BDF-CF methods for the incompressible Navier-Stokes. The Taylor vortex problem with $Re = 2\pi^2$ considered. We use Dirichlet BCs on $\Omega = [-1, 1]^2$ and SEM of order $N = 12$ with $Ne = 4$ uniform rectangular elements. Errors are measured at time $T = 1$ and plotted as functions of time step $h = T/2^r$, $r = 4, \dots, 9$. Methods of order 1, 2, 3 are represented by circles (\circ), squares (\square) and triangles (∇) respectively. **(a)** The errors in the velocity measured in the H_1 -norm. **(b)** The errors in the pressure measured in the L_2 -norm.

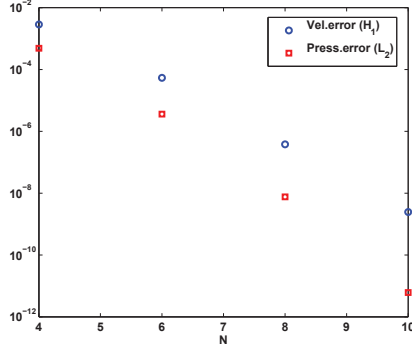


Figure 4.3: Spectral convergence in space, verified using BDF4-CF with time step $h = T/2^8$. The Taylor vortex problem with $Re = 2\pi^2$ considered. We use Dirichlet BCs on $\Omega = [-1, 1]^2$ and SEM with $Ne = 4$ uniform rectangular elements. Errors are measured at time $T = 1$ and plotted as functions spectral order $N = 4, 6, 8, 10$, on a semi-log scale. Errors in the velocity are measured in the H_1 -norm and plotted using circles (\circ) while errors in the pressure are measured in the L_2 -norm and plotted using squares (\square).

4.4 Stability of the BDF-CF methods

We study the stability of the BDF-CF methods, and make some comparisons with the IMEX multistep (SBDF) methods of Ascher *et al.* [4], also studied in [21]. The following remark shows a relation between the BDF-CF methods and the SBDF methods.

Remark 4.4.1. If we introduce linearizations of the form

$$\exp(hC(y_0))y_1 \approx y_1 + hC(y_0)y_1$$

in the BDF-CF1, BDF-CF2 (with $\gamma = 0$) and BDF-CF3 (with $\alpha = 0, \beta = 2, \gamma = 1$), we obtain exactly the SBDF methods of Ascher *et al.* [4].

4.4.1 A nonlinear problem

The authors in [4] demonstrated the strong stability and time-step restrictions of the SBDF methods among others, in the treatment of convection-diffusion problems with small viscosity coefficients. An interesting observation is the improved stability of the BDF-CF over the SBDF methods at smaller viscosities. We consider the Burgers' equation in 1D

$$u_t + uu_x = \nu u_{xx}, \quad x \in (-1, 1), \quad t \in (0, T], \quad (4.4.1)$$

with initial condition $u(0, x) = \sin \pi x$, and homogeneous Dirichlet boundary conditions. We discretize in space via the Gauss-Lobatto-Chebyshev spectral collocation method to

obtain an ODE of the form (4.2.1). This same test problem was considered in [4]. In Figure 4.4 we show the relative error⁵ in L_∞ grid-norm measured at time $T = 2$ for a range of viscosity parameters in the range $0.001 \leq \nu \leq 0.1$. For each time step $h = 1/10, 1/20, 1/40, 1/80$, we have used $N = 40$ spatial nodes. The reference or “exact” solution is computed for $N = 80$ spatial nodes using MATLAB’s `ode45` built in function, with sufficiently small relative and absolute error tolerances. We have used `ode45` to obtain the starting values of each of the multistep methods. For BDF2-CF we used $\gamma = -1$, while for BDF3-CF we used $\alpha = 1, \beta = -13/2, \gamma = 3$.

An observation from Figure 4.4 reveals that the BDF-CF methods are numerical more stable (and allow for larger time steps) than the SBDF methods. Unlike the BDF-CF methods, the SBDF methods give unbounded solutions at smaller viscosity parameters, especially as the Courant number increases with increasing time step h . The better performance of the BDF-CF methods at low viscosities is believed to be due to the (accurate) semi-Lagrangian computation of exponential flows (see also [7]).

4.4.2 Linear Stability

We now consider a linear stability analysis like the one done in [4], whereby we apply the methods to a simple problem of the type

$$\dot{y} = (\lambda + \hat{v})y, \quad (4.4.2)$$

where $\lambda, \nu \in \mathbb{R}$, and \hat{v} is the unit imaginary number satisfying $\hat{v}^2 = -1$. Notice that (4.4.2) is equivalent to (4.2.1) with $C(y) = \hat{v}y$ and $f(y) = \lambda y$, where I denote the identity matrix.

Let $\omega := (\lambda + \hat{v})h \in \mathbb{C}$, and let ω_r and ω_i denote the real and imaginary parts of ω respectively, suppressing the dependence on h . Applying the SBDF2 method to (4.4.2) yields the characteristic polynomial

$$\Phi(\tau; \omega) := (3 - 2\omega_r)\tau^2 - 4(1 + \hat{v}\omega_i)\tau + 2\hat{v}\omega_i + 1;$$

meanwhile the BDF2-CF method has characteristic polynomial

$$\Phi(\tau; \omega) := (3 - 2\omega_r)\tau^2 - 4e^{i\omega_i}\tau + e^{2i\omega_i}.$$

The stability region is then given by the set

$$\mathcal{S} := \{\omega \in \mathbb{C} : |\max\{\tau : \Phi(\tau; \omega) = 0\}| \leq 1\}$$

In Table 4.1 we write down the characteristic polynomials of the second to fourth order BDF-CF and SBDF methods. Figure 4.5 shows the stability regions (shaded gray) for these SBDF and BDF-CF methods. Also included in this figure is a comparison with some second, third and fourth order methods of Hundsdorfer and Ruuth [21], namely, IMEX-Shu (3,2), IMEX-TVB_o(3,3), and IMEX-TVB(4,4). The latter methods have been shown [21] to have very good linear stability and optimal damping properties.

⁵relative to the exact solution

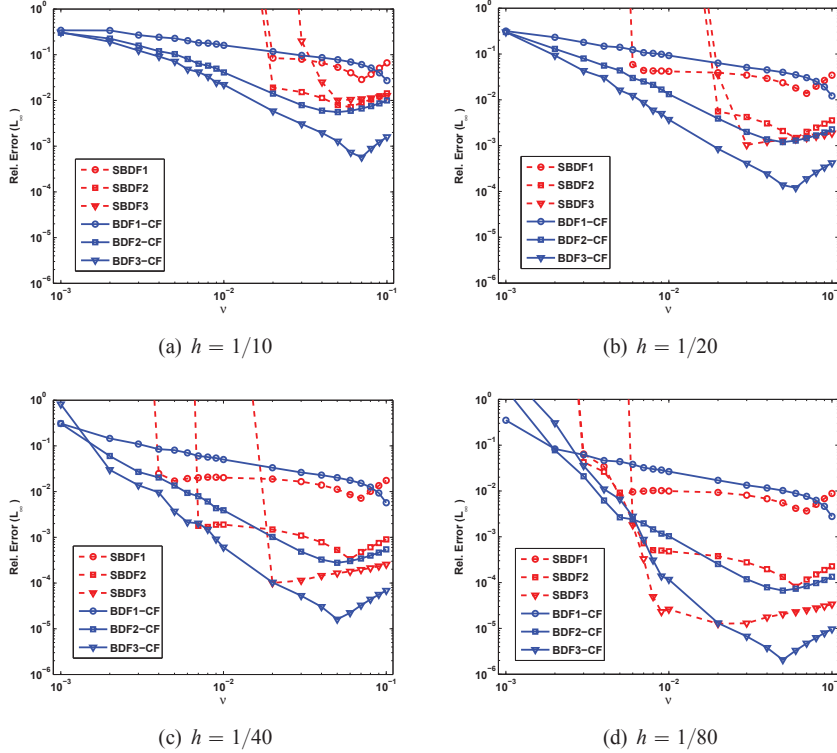


Figure 4.4: Burgers' equation over a range of viscosity parameters. We use Dirichlet BCs on the domain $[-1, 1]$; $N = 40$. Relative errors (in the L_∞ -norm) are plotted at time $T = 2$ as functions of viscosity $\nu \in \{0.001, 0.002, \dots, 0.009, 0.01, 0.02, \dots, 0.1\}$, for time steps $h = 1/10, 1/20, 1/40, 1/80$. SBDFk and BDFk-CF represent k order methods. SBDF methods are represented by dashed (---) lines, and BDF-CF methods by solid lines. Methods of order $k = 1, 2, 3$ are represented by circles (\circ), squares (\square) and triangles (∇) respectively

We observe from Figure 4.5 that all the BDF-CF methods are A -stable. This is clearly not the case with the other methods. In particular the BDF2-CF has characteristic roots given by

$$\tau_{1,2} = e^{i\omega_i} [2 \pm \sqrt{1 + 2\omega_r}] / (3 - 2\omega_r),$$

and it is easy to show that for $\omega_r \leq 0$ we have $|2 \pm \sqrt{1 + 2\omega_r}| \leq |3 - 2\omega_r|$, which implies that $|\tau_{1,2}| \leq 1$. In fact, given $\omega_r = -r$, $r \geq 0$,

$$|2 \pm \sqrt{1 + 2\omega_r}| = |2 \pm \sqrt{1 - 2r}| \leq 2 + \sqrt{|1 - 2r|}.$$

Table 4.1: Characteristic polynomials for BDF-CF and SBDF methods

order	BDF-CF
2	$(\frac{3}{2} - \omega_r) \tau^2 - 2e^{i\omega_i} \tau + \frac{1}{2}e^{2i\omega_i}$
3	$(\frac{11}{6} - \omega_r) \tau^3 - 3e^{i\omega_i} \tau^2 + \frac{3}{2}e^{2i\omega_i} \tau - \frac{1}{3}e^{3i\omega_i}$
4	$(\frac{25}{12} - \omega_r) \tau^4 - 4e^{i\omega_i} \tau^3 + 3e^{2i\omega_i} \tau^2 - \frac{4}{3}e^{3i\omega_i} + \frac{1}{4}e^{4i\omega_i}$
order	SBDF
2	$(\frac{3}{2} - \omega_r) \tau^2 - 2(1 + i\omega_i)\tau + \frac{1}{2}(1 + 2i\omega_i)$
3	$(\frac{11}{6} - \omega_r) \tau^3 - 3(1 + i\omega_i)\tau^2 + \frac{3}{2}(1 + 2i\omega_i)\tau - \frac{1}{3}(1 + 3i\omega_i)$
4	$(\frac{25}{12} - \omega_r) \tau^4 - 4(1 + i\omega_i)\tau^3 + 3(1 + 2i\omega_i)\tau^2 - \frac{4}{3}(1 + 3i\omega_i) + \frac{1}{4}(1 + 4i\omega_i)$

For $0 \leq r \leq 1/2$,

$$2 + \sqrt{|1 - 2r|} = 2 + \sqrt{1 - 2r} \leq 3 \leq 3 + 2r = |3 - 2\omega_r|;$$

and for $r > 1/2$,

$$2 + \sqrt{|1 - 2r|} = 2 + \sqrt{2r - 1} \leq 2 + 2r \leq 3 + 2r = |3 - 2\omega_r|.$$

On the other hand, the SBDF and IMEX-TVb methods are only $A(\alpha)$ -stable, in the sense of [17, Definition 2.1, p.250], with $0 < \alpha < 90^\circ$ (see Figure 4.5). The SBDF4 shows even smaller stability region than the methods of lower order in its class.

We can thus conclude that for a linear convection-diffusion problem with constant coefficients and diffusion parameter $\nu > 0$, the time-step restrictions due to stability are much more relaxed in the BDF-CF methods than for the SBDF, both in the cases of small and large ν . The SBDF methods pose even more severe time-step restrictions for small ν . This is evident from the stability regions plotted in Figure 4.5.

Conclusion

We have derived new exponential multistep methods based on the BDF schemes. The methods can achieve order of convergence higher than 2 when applied to index 2 DAEs. The methods have good linear stability properties. The numerical experiments reveal that the methods are suitable for convection-dominated convection-diffusion problems when implemented in a semi-Lagrangian fashion.

The framework we propose does not depend on the type of spatial discretization involved, however it is important to choose spatial discretization methods that would give rise to ODEs/DAEs with large but sparse convection operators, because this will ease the computation of the exponential flows.

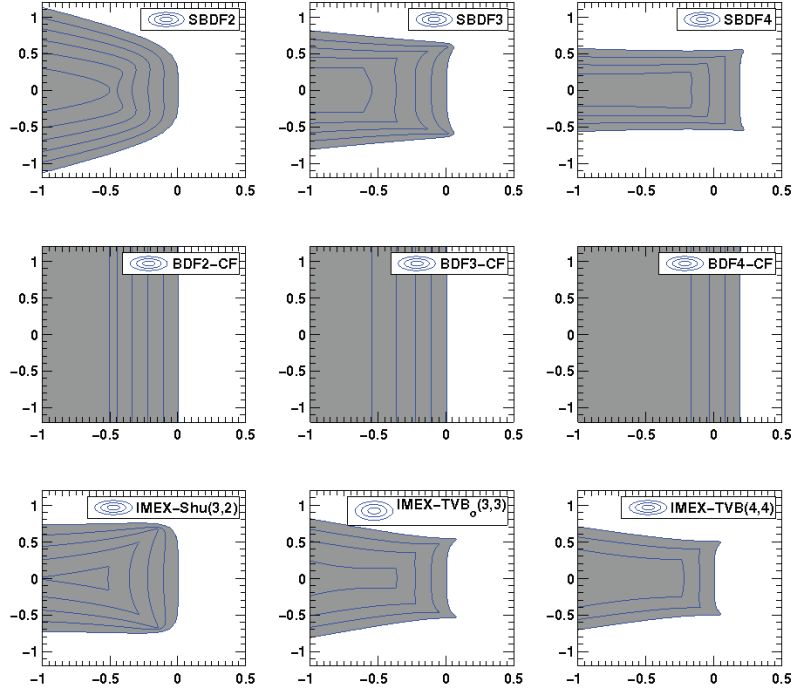


Figure 4.5: Stability domains $\mathcal{S} \subset \mathbb{C}$ (shaded gray) for some SBDF and BDF-CF methods. Also included (third row) are some IMEX multistep methods by Hundsdorfer *et al.* [21]. Each domain \mathcal{S} (partly shown) is unbounded and includes the whole negative real axis, and part/whole of the imaginary axis.

For example when using the SEM or other domain decomposition strategies for the space discretization, one would distribute the work of computing the exponentials on different processors while storing the information relative to the convection operator locally. In our implementation we use a semi-Lagrangian approach, adopting the techniques of [41] for the computation of the characteristics.

It is evident that standard IMEX methods require a smaller cost per step as the convection is treated explicitly. On the other hand a proper comparison of IMEX versus semi-Lagrangian methods should take into account that the latter allow the use of larger time steps, at least for certain viscosity ranges. Our Navier-Stokes experiments are based on the SEM spatial discretization, and we have not yet explored all the possibilities for an

efficient approximation of the exponentials in this setting. To draw definite conclusions about this comparison is premature at this stage.

The convergence of the methods has been verified numerically on a Navier-Stokes problem. An interesting future work will be to analyze or investigate the Courant-Friedrichs-Lewy (CFL) condition in the limiting case as the viscous term vanishes.

Acknowledgements

Our appreciations to Einar Rønquist (Mathematics department, NTNU, Norway) for his invaluable insights on the use spectral element methods in the discretization of Navier-Stokes equations. Further appreciations go to Tormod Bjøntegaard and Olivier Verdier (Mathematics department, NTNU, Norway) for fruitful discussions. Sincere thanks also go to the members of the numerical analysis group at NTNU.

4.5 Appendix

4.5.1 Definition of norms

For a square-integrable (respectively H_1) function $\mathbf{u} : \Omega \rightarrow \mathbb{R}^n$, where $\Omega \subset \mathbb{R}^m$ is bounded and connected, the L_2 -norm ($\|\cdot\|_{L_2}$) and the H_1 -norm ($\|\cdot\|_{H_1}$) are defined by

$$\|\mathbf{u}\|_{L_2} := \left\{ \sum_{i=1}^n \int_{\Omega} u_i^2 d\Omega \right\}^{1/2}, \quad (4.5.1)$$

$$\|\mathbf{u}\|_{H_1} := \left\{ \sum_{i=1}^n \int_{\Omega} (u_i^2 + \nabla u_i \cdot \nabla u_i) d\Omega \right\}^{1/2}. \quad (4.5.2)$$

In the spectral element approximations the continuous integrals of numerical solutions are accurately computed using Gauss quadrature rules.

Given a vector $y = (y_1, \dots, y_K)^T \in \mathbb{R}^K$, the *discrete* L_2 -norm ($\|\cdot\|_2$) and the L_∞ grid-norm ($\|\cdot\|_\infty$) are defined by

$$\|y\|_2 := \left\{ \sum_{j=1}^K y_j^2 \right\}^{1/2}, \quad \|y\|_\infty := \max_{1 \leq j \leq K} |y_j|. \quad (4.5.3)$$

4.5.2 Order conditions for order 3 method BDF3-CF

$$\begin{aligned} 3(a_{31} + a_{32} + a_{33}) - \frac{3}{2}(a_{21} + a_{22} + a_{23}) + \frac{1}{3}(a_{11} + a_{12} + a_{13}) &= 1, \\ 3(a_{31} + a_{32} + a_{33}) - \frac{1}{3}(a_{11} + a_{12} + a_{13}) &= 2, \end{aligned}$$

$$\begin{aligned}
 3(a_{33} - a_{31}) - \frac{3}{2}(a_{23} - a_{21}) + \frac{1}{3}(a_{13} - a_{11}) &= 2, \\
 3(a_{31} + a_{32} + a_{33}) + \frac{1}{3}(a_{11} + a_{12} + a_{13}) &= 4, \\
 3(a_{31} + a_{33}) - \frac{3}{2}(a_{21} + a_{23}) + \frac{1}{3}(a_{11} + a_{13}) &= 4, \\
 3(a_{33} - a_{31}) - \frac{1}{3}(a_{13} - a_{11}) &= 4, \\
 3(a_{31} + a_{32} + a_{33})^2 - \frac{3}{2}(a_{21} + a_{22} + a_{23})^2 + \frac{1}{3}(a_{11} + a_{12} + a_{13})^2 &= 0, \\
 3(a_{31} + a_{32} + a_{33})^2 - \frac{1}{3}(a_{11} + a_{12} + a_{13})^2 &= 0, \\
 3(a_{31} + a_{32} + a_{33})^3 - \frac{3}{2}(a_{21} + a_{22} + a_{23})^3 + \frac{1}{3}(a_{11} + a_{12} + a_{13})^3 &= 0, \\
 3(a_{31} + a_{32} + a_{33})(a_{33} - a_{31}) - \frac{3}{2}(a_{21} + a_{22} + a_{23})(a_{23} - a_{21}) \\
 + \frac{1}{3}(a_{11} + a_{12} + a_{13})(a_{13} - a_{11}) &= 0.
 \end{aligned}$$

4.5.3 DIRK-CF methods and commutator-free methods

We consider vector fields F on \mathbb{R}^d , to be smooth functions assigning to each point $y \in \mathbb{R}^d$ a tangent vector at y . The vector field F can be expressed in coordinates as

$$F(y) = \sum_{i=1}^d f_i(y) \left. \frac{\partial}{\partial y_i} \right|_y,$$

to emphasize the fact that $F(y)$ is a tangent vector [39], $f_i(y)$ are the components of the tangent vector and $\left. \frac{\partial}{\partial y_i} \right|_y$ is the canonical basis of the tangent space to \mathbb{R}^d at y .

The flow at time t of F through the point y_0 is denoted by

$$y(t) = \exp(tF)y_0, \quad (4.5.4)$$

where $y(t)$ satisfies the differential equation

$$\dot{y}_i(t) = f_i(y(t)), \quad i = 1, \dots, d, \quad y(t_0) = y_0.$$

The Lie bracket of two vector fields, F and G (or commutator) is a third vector field obtained by applying first F to G as a differential operator, and then subtracting the result of applying G to F . This leads to

$$[F, G] := \sum_{j=1}^d \sum_{i=1}^d \left(f_i \frac{\partial g_j}{\partial y_i} - g_i \frac{\partial f_j}{\partial y_i} \right) \frac{\partial}{\partial y_j}.$$

For vector fields F and G such that

$$\begin{bmatrix} f_1(y) \\ \vdots \\ f_d(y) \end{bmatrix} = Ay, \quad \begin{bmatrix} g_1(y) \\ \vdots \\ g_d(y) \end{bmatrix} = By$$

where A and B are $d \times d$ matrices, the Lie bracket of vector fields is the vector field $[F, G]$ with components Cy and $C = BA - AB$ is the matrix commutator of B and A . The composition of the flows of two vector fields F and G can be expressed as the flow of a third vector field defined by means of a series of iterated Lie brackets of the two vector fields F and G .

The set of vector fields on \mathbb{R}^d , written $\mathcal{X}(\mathbb{R}^d)$, forms a Lie algebra.

A set of vector fields $\{\mathcal{E}_1, \dots, \mathcal{E}_m\}$, $d \leq m$, is a set of *frame vector fields* on \mathbb{R}^d if

$$\mathbb{R}^d = \text{span}\{\mathcal{E}_1|_x, \dots, \mathcal{E}_m|_x\}, \quad \forall x \in \mathbb{R}^d.$$

Given any vector field $F \in \mathcal{X}(\mathbb{R}^d)$ we have

$$F(y) = \sum_{i=1}^m f_i(y) \mathcal{E}_i(y).$$

We denote by F_p the vector field

$$F_p(x) = \sum_{i=1}^m f_i(p) \mathcal{E}_i(x)$$

and we say that F_p is the vector field F *frozen* at the point p .

We define Runge-Kutta commutator-free methods [12, 8] for approximating the flow (4.5.4) as follows:

Algorithm 4.2. *Commutator-free method*

$p = y_n$

for $r = 1 : s$ **do**

$$Y_r = \exp\left(\sum_{k=1}^s \alpha_{r,k}^k F_k\right) \cdots \exp\left(\sum_{k=1}^s \alpha_{r,1}^k F_k\right) p$$

$$F_r = hF_{Y_r} = h \sum_{i=1}^m f_i(Y_r) \mathcal{E}_i$$

end

$$y_{n+1} = \exp\left(\sum_{k=1}^s \beta_k^k F_k\right) \cdots \exp\left(\sum_{k=1}^s \beta_1^k F_k\right) p$$

Chapter 4. Semi-Lagrangian multistep exponential integrators for index 2 DAEs

Here n counts the number of time steps and h is the step-size of integration. The integrator has s stages and parameters $\alpha_{rl}^k, \beta_l^k, r, k = 1, \dots, s$ and $l = 1, \dots, J$. Each new stage value is obtained as a composition of J exponentials (i.e. exact flows) of linear combinations of vector fields frozen at the previously computed stage values. In the case $\mathcal{E}_i(y) = \frac{\partial}{\partial y_i}$ the commutator-free methods reduce to the usual Runge-Kutta methods and the exponentials are simple translations in \mathbb{R}^d . Note that composition of exponentials can be replaced by truncated series of repeated commutators, but this is not always preferable from the point of view of computational complexity.

$$\begin{bmatrix} f_1(y) \\ \vdots \\ f_d(y) \end{bmatrix} = Ay + C(y)y.$$

We denote by IMEX-CF an IMEX Runge-Kutta method where the implicit part is a usual implicit Runge-Kutta method applied to Ay and the explicit part is a commutator-free method applied to $C(y)y$. We choose the set of frame vector fields $\mathcal{E}_i(y)$ with components $E_i y$, where $E_i, i = 1, \dots, d^2$, is any fixed basis of the vector space of $d \times d$ matrices. This implies that the vector field corresponding to $C(y)y$ frozen at p has components $C(p)y$. The IMEX-CF method has the form reported in Algorithm 4.3 (see e.g.[7]). The method is easy to implement when the coefficients $\{a_{ij}, b_i, c_i\}, i, j = 1, \dots, s$, define a DIRK method. In this case we denote the method by DIRK-CF. When applied to problems of the type (4.1.1) the constraints, $g(Y_i) = 0, g(y_{n+1}) = 0$, could be enforced via a projection technique as in [30, 2, 29, 17].

Algorithm 4.3. IMEX-CF method

for $i = 1$ **to** s **do**
 $\varphi_i = \exp(h \sum_k \alpha_{iJ}^k C(Y_k)) \cdot \dots \cdot \exp(h \sum_k \alpha_{i1}^k C(Y_k)),$
 $Y_i = \varphi_i y_n + h \sum_j a_{ij} \varphi_i \varphi_j^{-1} A Y_j,$
end
 $\varphi_{s+1} = \exp(h \sum_k \beta_J^k C(y_k)) \cdot \dots \cdot \exp(h \sum_k \beta_1^k C(y_k)),$
 $y_{n+1} = \varphi_{s+1} y_n + h \sum_i b_i \varphi_{s+1} \varphi_i^{-1} A Y_i,$

References

- [1] R. Alexander, *Diagonally Implicit Runge-Kutta methods for stiff o.d.e.'s*, SIAM J. Numer. Anal. **14** (1977), no. 6, 1006–1021.

-
- [2] U. M. Ascher and L. R. Petzold, *Projected implicit Runge-Kutta methods for differential-algebraic equations*, SIAM J. Numer. Anal. **28** (1991), no. 4, 1097–1120.
- [3] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri, *Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations*, Appl. Numer. Math. **25** (1997), 151–167.
- [4] U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton, *Implicit-explicit methods for time-dependent partial differential equations*, SIAM J. Numer. Anal. **32** (1995), no. 3, 797–823.
- [5] M. P. Calvo and C. Palencia, *A class of explicit multistep exponential integrators for semilinear problems*, Numer. Math. **102** (2006), no. 3, 367–381.
- [6] E. Celledoni, *Eulerian and semi-Lagrangian commutator-free exponential integrators*, CRM Proceedings and Lecture Notes **39** (2005), 77–90.
- [7] E. Celledoni and B. K. Kometa, *Order conditions for semi-Lagrangian Runge-Kutta exponential integrators*, Preprint series: Numerics, 04/2009, Department of Mathematics, NTNU, Trondheim, Norway, <http://www.math.ntnu.no/preprint/numerics/N4-2009.pdf>, 2009.
- [8] E. Celledoni, A. Marthinsen, and B. Owren, *Commutator-free Lie group methods*, FGCS **19** (2003).
- [9] E. Celledoni and I. Moret, *A Krylov projection method for systems of ODEs*, Appl. Numer. Math. **24** (1997), no. 2-3, 365–378, Volterra centennial (Tempe, AZ, 1996). MR 1464736 (98g:65061)
- [10] E. M. Constantinescu and A. Sandu, *Extrapolated implicit-explicit time stepping*, SIAM J. Sci. Comput. **31** (2009/10), no. 6, 4452–4477. MR 2594989 (2011d:65166)
- [11] S. M. Cox and P. C. Matthews, *Exponential time differencing for stiff systems*, J. Comput. Phys. **176** (2002), no. 2, 430–455. MR 1894772 (2003b:65064)
- [12] P. E. Crouch and R. Grossman, *Numerical integration of ordinary differential equations on manifolds*, J. Nonlinear Sci. **3** (1993), no. 1, 1–33. MR 1216986 (94e:65069)
- [13] P. F. Fischer, *An overlapping Schwarz method for spectral element solution of the incompressible Navier-Stokes equations*, J. Comput. Phys. **133** (1997), no. 1, 84–101. MR 1445173 (97m:76094)
- [14] P. F. Fischer, G. W. Kruse, and F. Loth, *Spectral element method for transitional flows in complex geometries*, J. Sc. Comput. **17** (2002), no. 1–4, 81–98.
- [15] F. X. Giraldo, *The Lagrange-Galerkin spectral element method on unstructured quadrilateral grids*, J. Comput. Phys. **147** (1998), no. 1, 114–146. MR 1657765 (99j:65177)

Chapter 4. Semi-Lagrangian multistep exponential integrators for index 2 DAEs

- [16] E. Hairer, Ch. Lubich, and M. Roche, *The numerical solution of differential-algebraic systems by Runge-Kutta methods*, Lecture Notes in Mathematics, vol. 1409, Springer-Verlag, Berlin, 1989.
- [17] E. Hairer and G. Wanner, *Solving ordinary differential equations. II*, second ed., Springer Series in Computational Mathematics, vol. 14, Springer-Verlag, Berlin, 1996, Stiff and differential-algebraic problems.
- [18] I. Higuera and T. Roldán, *Starting algorithms for a class of RK methods for index-2 DAEs*, *Comput. Math. Appl.* **49** (2005), no. 7-8, 1081–1099.
- [19] M. Hochbruck and Ch. Lubich, *On Krylov subspace approximations to the matrix exponential operator*, *SIAM J. Numer. Anal.* **34** (1997), no. 5, 1911–1925. MR 1472203 (98h:65018)
- [20] S. Hugues and A. Randriamampianina, *An improved projection scheme applied to pseudospectral methods for the incompressible Navier-Stokes equations*, *Internat. J. Numer. Methods Fluids* **28** (1998), no. 3, 501–521.
- [21] W. Hundsdorfer and S. J. Ruuth, *IMEX extensions of linear multistep methods with general monotonicity and boundedness properties*, *J. Comput. Phys.* **225** (2007), no. 2, 2016–2042.
- [22] L. O. Jay, *Convergence of a class of Runge-Kutta methods for differential-algebraic systems of index 2*, *BIT* **33** (1993), no. 1, 137–150.
- [23] ———, *Solution of index 2 implicit differential-algebraic equations by Lobatto Runge-Kutta methods*, *BIT* **43** (2003), no. 1, 93–106.
- [24] ———, *Specialized Runge-Kutta methods for index 2 differential-algebraic equations*, *Math. Comp.* **75** (2006), no. 254, 641–654 (electronic).
- [25] G. E. Karniadakis, M. Israeli, and S. A. Orszag, *High-order splitting methods for the incompressible Navier-Stokes equations*, *J. Comput. Phys.* **97** (1991), no. 2, 414–443.
- [26] C. A. Kennedy and M. H. Carpenter, *Additive Runge-Kutta schemes for convection-diffusion-reaction equations*, *Appl. Numer. Math.* **44** (2003), no. 1-2, 139–181.
- [27] A. Kværnø, *Singly diagonally implicit Runge-Kutta methods with an explicit first stage*, *BIT* **44** (2004), no. 3, 489–502.
- [28] L. Lopez and V. Simoncini, *Analysis of projection methods for rational function approximation to the matrix exponential*, *SIAM J. Numer. Anal.* **44** (2006), no. 2, 613–635 (electronic). MR 2218962 (2007b:65036)
- [29] Ch. Lubich, *On projected Runge-Kutta methods for differential-algebraic equations*, *BIT* **31** (1991), no. 3, 545–550.
- [30] Y. Maday, A. T. Patera, and E. M. Rønquist, *An operator-integration-factor splitting*

-
- method for time-dependent problems: application to incompressible fluid flow*, J. Sci. Comput. **5** (1990), no. 4, 263–292.
- [31] I. Moret and P. Novati, *RD-rational approximations of the matrix exponential*, BIT **44** (2004), no. 3, 595–615. MR 2106019 (2005h:65073)
- [32] A. Ostermann and M. Thalhammer, *Positivity of exponential multistep methods*, Numerical mathematics and advanced applications, Springer, Berlin, 2006, pp. 564–571.
- [33] L. Pareschi and G. Russo, *Implicit-Explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation*, J. Sci. Comput. **25** (2005), no. 1-2, 129–155. MR 2231946 (2007b:65063)
- [34] R. Peyret, *Spectral methods for incompressible viscous flow*, Appl. Math. Sc., vol. 148, Springer, Berlin, 2000.
- [35] O. Pironneau, *On the transport-diffusion algorithm and its applications to the Navier-Stokes equations*, Numer. Math. **38** (1981/82), no. 3, 309–332. MR 654100 (83d:65258)
- [36] J. Rang, *Design of DIRK schemes for solving Navier-Stokes equations*, Informatik-bericht 2007-02, Technische Universität Braunschweig (2007), no. 2.
- [37] K. Shahbazi, P. F. Fischer, and C. R. Ethier, *A high-order discontinuous Galerkin method for the unsteady incompressible Navier-Stokes equations*, J. Comput. Phys. **222** (2007), no. 1, 391–407.
- [38] R. B. Sidje, *Expokit: a software package for computing matrix exponentials*, ACM Trans. Math. Softw. **24** (1998), 130–156.
- [39] L. W. Tu, *An introduction to manifolds*, second ed., Universitext, Springer, New York, 2011. MR 2723362 (2011g:58001)
- [40] J. Wensch, O. Knöth, and A. Galant, *Multirate infinitesimal step methods for atmospheric flow simulation*, BIT **49** (2009), no. 2, 449–473.
- [41] D. Xiu and G. E. Karniadakis, *A semi-Lagrangian high-order method for Navier-Stokes equations*, J. Comput. Phys. **172** (2001), no. 2, 658–684. MR 1857617 (2002g:76077)

Paper IV

**On discontinuous Galerkin methods and
commutator-free exponential integrators for
advection problems**

Bawfeh Kingsley Kometa

Submitted to *Proceedings of the ICNAAM conference, 2011*

Chapter 5

On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems

Abstract. Discontinuous Galerkin (DG) finite-element methods are well-known to be suitable for solving convection-dominated convection diffusion problems. High order Runge-Kutta methods such as the RKDG and SSP methods have been developed and tested to work quite well for convection-dominated problems (see e.g., Cockburn & Shu (2001), Gottlieb *et al.* (2001,2009)). An issue of concern remains the strong CFL restrictions on the discretization parameters. Restelli *et al.* (2006) proposed combining the DG methods with the semi-Lagrangian methods in what they called *semi-Lagrangian discontinuous Galerkin* (SLDG) methods. Proposed implementations of the SLDG methods however suffer from limited spatial accuracy as opposed to the RKDG methods. We hereby propose a method in the framework of Lie-group exponential integrators (see Cellodoni *et al.* [6, 5]), that uses a modified version of the SLDG method as a building block for computing compositions of convection flows and maintain the good properties of the DG formulations.

5.1 Introduction

We study advection problems of the form

$$u_t + \mathbf{V} \cdot \nabla u = 0, \tag{5.1.1}$$

Chapter 5. On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems

where $u = u(\mathbf{x}, t) \in \mathbb{R}$ is an unknown scalar field dependent on space and time variables \mathbf{x} and t respectively, while (in a more general setting) $\mathbf{V} = \mathbf{V}(u, \mathbf{x}, t) \in \mathbb{R}^d$, $d = 1, 2$ or 3 , represents a given advection velocity. The equation is treated over a bounded uniform domain $(\mathbf{x}, t) \in \Omega \times (0, T) \subset \mathbb{R}^d \times \mathbb{R}$ with suitably prescribed initial and boundary conditions on u . We denote by subscript- t the partial derivative with respect to time; $\nabla \cdot$ is the divergence operator with respect to \mathbf{x} . Equation (5.1.1) is said to be in *advective or Lagrangian form*. Throughout the rest of the paper we assume that the advection velocity \mathbf{V} is divergence-free (i.e. $\nabla \cdot \mathbf{V} = 0$). Under this assumption (5.1.1) becomes equivalent to the *conservative form*

$$u_t + \nabla \cdot (\mathbf{V}u) = 0. \quad (5.1.2)$$

The form (5.1.2) is suitable for formulating discontinuous Galerkin methods, while (5.1.1) allows for the use of traditional semi-Lagrangian methods.

Discontinuous Galerkin (DG) finite-element methods are well-known to be suitable for solving convection-dominated convection diffusion problems. This is due to their ability to admit solution profiles with jump or contact discontinuities. High order methods such as the Runge-Kutta discontinuous Galerkin (RKDG) and the strong stability-preserving (SSP) methods have been developed and tested to work quite well for the treatment of hyperbolic conservation laws (see e.g., [8, 12, 11]). However the CFL restriction on the discretization parameters is still an issue of concern especially for methods with high temporal order. Nevertheless, the high spatial accuracy and high level of parallelism of these methods remain attractive from a numerical point of view. Semi-Lagrangian (SL) methods on the other hand are well known to be very efficient and accurate (see for example [15, 9] and references therein). The SLDG methods of Restelli *et al.* [19] aim at combining the good properties of both the SL and DG methods. Such methods were found useful for applications in nonhydrostatic atmospheric modelling. Implementations of the SLDG methods however suffer from low spatial accuracy as opposed to RKDG methods.

In this paper we study the relations between commutator-free Lie-group exponential integrators (CF) of Celledoni *et al.* [6] and the semi-Lagrangian discontinuous Galerkin methods (SLDG) presented in [19] for advection problems. We reformulate the CF methods using the SLDG method as a building block for computing the pure convection flows. When compared to RKDG methods of high temporal order, the new formulation allow for the use of larger Courant numbers. The spatial discretization is based on high order Gauss-Lobatto-Legendre (GLL) polynomial approximations. The SLDG component is thus modified to maintain this high spatial accuracy. Numerical experiments are presented both for 1D and 2D advection problems to demonstrate the performance of the new methods.

5.2 Discontinuous Galerkin methods for hyperbolic conservation laws

Following the recipes of Cockburn and Shu [8] for the spatial discretization of (5.1.2) one obtains a semi-discrete system of ODEs.

$$u_t + \nabla \cdot f(u) = 0, \quad \mathbf{x} \in \Omega, \quad t > 0, \quad (5.2.1)$$

with prescribed boundary conditions and initial data u_0 . We observe that $f(u) := \mathbf{V}u$, in relation to (5.1.2).

5.2.1 Weak formulation on a broken Sobolev space

Consider a finite element discretization of (5.2.1) in which Ω is subdivided into N_e subdomains (or “elements”) denoted by $K := \Omega_j$, $j = 1, \dots, N_e$. Further assume that the functions u and $f(u) = (f \circ u)(\mathbf{x})$ are smooth on Ω . We multiply (5.2.1) by a test function v and integrate by parts over a subdomain K to obtain

$$\int_K u_t v, d\mathbf{x} - \int_K f(u) \cdot \nabla v d\mathbf{x} + \int_{\partial K} f(u) \cdot \mathbf{n}_K v ds = 0, \quad (5.2.2)$$

where \mathbf{n}_K is the outward unit normal on the subdomain boundary ∂K . The goal, however, is to allow for functions u that admit jump discontinuities at the element boundaries ∂K , and have a certain amount of regularity within the interior of each element. The space described by such functions is referred to as the ‘*broken*’ Sobolev space in the DG literature (see for example [16, 2, 13]).

Now let u be a function in a broken Sobolev space $H^1 := H^1(\Omega)$ (functions with H_1 regularity in each element K). Note that for such u the flux function $f(u)$ is well-defined on Ω *except* (possibly) at the element boundaries, where u may have a jump discontinuity. To remove this ambiguity the restriction of $f(u)$ on the nodal points is replaced by a function $\hat{f}(u)$, also known as the *numerical flux function*. The function $\hat{f}(u)$ can be seen as a well-defined (single-valued) approximation to $f(u)$. It is defined such that

$$\hat{f}(u)|_{\partial K} := \hat{f}(u^-, u^+), \quad \text{and} \quad \hat{f}(u)|_K := f(u), \quad (5.2.3)$$

where for each $s \in \partial K$, $u^+(s)$ denotes the value of the trace of u on K and $u^-(s)$ denotes the value of the trace of u on the neighbouring element K^- having $s \in \partial K^-$. The function $\hat{f}(\cdot, \cdot)$ can be defined in several ways. For example, in the case of locally conservative schemes, it can represent a Godunov, Lax-Friedrichs, Ösher-Enquist, or an upwind flux function (see [8]). Two important requirements for the definition of $\hat{f}(\cdot, \cdot)$ include

- (a) Consistency: $\hat{f}(v, v) = f(v)$, for any v ;
- (b) Monotonicity: $\hat{f}(v, \cdot)$ is non-decreasing, while $\hat{f}(\cdot, v)$ is non-increasing (a necessary requirement for constructing monotone schemes).

Chapter 5. On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems

Now summing (5.2.2) over all elements and introducing the numerical flux function at element boundaries, we get the following *weak formulation*:

Find $u \in H^1 := H^1(\Omega)$ such that, for $t > 0$,

$$\sum_K \int_K (u_t v - f(u) \cdot \nabla v) \, d\mathbf{x} + \sum_K \int_{\partial K} f(u) \cdot \mathbf{n}_K v \, ds = 0, \quad \text{for all } v \in H^1. \quad (5.2.4)$$

5.2.2 The discrete weak formulation on a piece-wise polynomial space

Let $\mathcal{T} := \mathcal{T}_h$ denote a triangulation of the domain Ω (i.e., the set of all elements of Ω) with a mesh parameter given by $h = \max_{K \in \mathcal{T}} \text{diam}(K)$ (or $h = \max_{K \in \mathcal{T}} |K|$ in 1D). A simple discrete finite dimensional subspace of the broken H^1 space is given by

$$V_h^p := \{v \in L^2 : v|_K \in \mathbb{P}^p(K), \forall K \in \mathcal{T}_h\},$$

i.e. V_h^p consist of piecewise polynomials of degree p , possibly discontinuous across subdomain boundaries. Unless, not obvious, we shall henceforth suppress the writing of the superscript p in V_h^p .

An approximation for u in the discrete subspace V_h is sought for via the discrete weak formulation (related to (5.2.4)), namely: Find $u_h \in V_h$ such that, for $t > 0$,

$$\sum_K \int_K (u_{h,t} v - f(u_h) \cdot \nabla v) \, d\mathbf{x} + \sum_K \int_{\partial K} \hat{f}(u_h) \cdot \mathbf{n}_K v \, ds = 0, \quad \text{for all } v \in V_h, \quad (5.2.5)$$

where $u_{h,t} := \frac{\partial u_h}{\partial t}$.

Basis functions for V_h can be chosen as functions $\varphi_K^m = \psi_K^m \cdot \chi_K$, $m = 0, \dots, p$, $K \in \mathcal{T}_h$ where χ_K denotes the characteristic function on K , and ψ_K^m , $m = 0, \dots, p$ represent the basis functions for $\mathbb{P}^p(\Omega_j)$. For example, affine map of Lagrange interpolation polynomials based on Gauss-Legendre (GL) nodes in a reference domain $\hat{\Omega}$.

The resulting semi-discrete system assembled over all elements, would give rise to a system of ODEs of the form¹

$$\dot{y} = C(y)y + \hat{r}(y), \quad (5.2.6)$$

where $y \in \mathbb{R}^{\mathcal{N}}$ (\mathcal{N} = computational degrees of freedom), and $C(y)$ is a matrix-valued. Meanwhile $\hat{r}(y)$ is a vector representing interelement contributions resulting from the numerical flux terms in (5.2.5), and $C(y)y$ represents elemental contributions. The representation in (5.2.6) is obtainable, for example, in the semi-discretization of the Burgers' equation.

Suitable and popular time integration schemes for solving such ODE system are the TVD methods of Cockburn and Shu [20, 8], also known as the *RKDG* method. Applying the Lie-group commutator-free exponential integrators (CF) [6] for such ODE system is

¹Here we have, for simplicity, suppressed the dependence of the solution $y = y(t)$ on mesh parameter h .

not immediate, due to the presence of an extra flux term. We therefore interpret the CF methods within the DG framework in a way that permits the evaluation of flows (“exponentials”) of convecting vector fields in a manner similar to the SLDG schemes [19].

5.3 Semi-Lagrangian DG schemes and Commutator-free exponential integrators

Our goal in this section is to derive a suitable Semi-Lagrangian scheme in the DG setting for numerically computing the flows (exponentials) of frozen vector fields in CF integrators [6].

5.3.1 The semi-Lagrangian discontinuous Galerkin (SLDG) methods

Suppose \mathcal{T}_h is a triangulation of Ω into elements K , and let V_h denote the corresponding DG FEM space on \mathcal{T}_h . Then the SLDG method for approximating the solution of (5.1.1) (or equivalently (5.1.2)) over one time step $[t_n, t_{n+1}]$, with step size $\Delta t := t_{n+1} - t_n$, follows the discrete weak formulation:

Given initial data $u_h^n \in V_h$, find $u_h^{n+1} \in V_h$ such that for each $K \in \mathcal{T}_h$

$$\begin{aligned} \int_K u_h^{n+1} v \, d\mathbf{x} &= \int_K u_h^n v \, d\mathbf{x} + \int_0^{\Delta t} \int_K [E_\tau^n u_h^n] \tilde{\mathbf{V}} \cdot \nabla v \, d\mathbf{x} d\tau \\ &\quad - \int_0^{\Delta t} \int_{\partial K} [E_\tau^n u_h^n] \tilde{\mathbf{V}} \cdot \mathbf{n}_K v \, ds d\tau, \quad \text{for all } v \in V_h, \end{aligned} \quad (5.3.1)$$

where $[E_\tau^n u_h^n](\mathbf{x})$ denotes the time *evolution operator* of u_h from time level t_n to time level $t_n + \tau$, defined such that

$$u_h(\mathbf{x}, t_n + \tau) = [E_\tau^n u_h^n](\mathbf{x}) := u_h(\chi(t_n + \tau), t_n),$$

χ being the solution of the ODE

$$\frac{d\chi(t)}{dt} = \tilde{\mathbf{V}}(\chi(t)), \quad t \in (t_n, t_n + \tau), \quad \text{given } \chi(t_n) = \mathbf{x}. \quad (5.3.2)$$

$\tilde{\mathbf{V}}$ is an approximation of \mathbf{V} that ensures the *numerical flux conservation*² across interelement boundaries. In order to achieve this, the authors of [19] proposed choosing $\tilde{\mathbf{V}}$ as the projection of \mathbf{V} on Raviart-Thomas FEM elements of lowest order. This however limits the spatial accuracy of the method. For the time integrals they used a quadrature with intermediate time nodes given such that $\tau \notin \{0, \Delta t\}$, e.g., the midpoint rule. This way any further ambiguity with evaluating the flux at interelement boundaries is avoided

²numerical fluxes are single-valued on interelement boundaries

Chapter 5. On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems

(at least for nonzero advecting velocity). A suitable numerical flux limiter is also used to ensure the monotonicity or positivity of the overall scheme.

The SLDG has been recently developed by Qiu and Shu [17] with essential modifications being the *exact* implementation of the evolution operator $E_\tau^n u_h^n$. Also the Divergence Theorem is used to evaluate the last two integrals in (5.3.1), and only Gauss quadratures and nodes are used (as opposed to Raviart-Thomas elements). The SLDG methods in [17] are further extended for the numerical treatment of the Vlasov-Poisson equations.

Inspired by pioneering work on DG methods for linear advection problems based on the *upwind principle* [18, 14] we make the following *modifications* to the SLDG (5.3.1): Find $u_h^{n+1} \in V_h$ such that for each element K ,

$$\begin{aligned} \int_K u_h^{n+1} v \, d\mathbf{x} &= \int_K u_h^n v \, d\mathbf{x} + \int_0^{\Delta t} \int_K [E_\tau^n u_h^n] \mathbf{V} \cdot \nabla v, \, d\mathbf{x} d\tau \\ &\quad - \int_0^{\Delta t} \int_{\partial K} [\widehat{E_\tau^n u_h^n}] \mathbf{V} \cdot \mathbf{n}_K v \, ds d\tau, \quad \text{for all } v \in V_h, \end{aligned} \quad (5.3.3)$$

where in $\widehat{(\cdot)}$ we consider the *upwind* values of u_h at element boundaries. The upwinding principle is also proposed in [17]. Other approximations for the numerical fluxes includes those mentioned in Section 5.2.1 and are generally acceptable. However, for the sake of simplicity, we use the upwinding principle. Also this choice, as opposed to low order Raviart-Thomas approximations, does not destroy the high spatial accuracy obtainable via the DG methods.

5.3.2 Commutator-free Lie group method

Semi-Lagrangian commutator-free Lie group methods (or exponential integrators) are known to be very accurate for approximating linear pure convection problems and have good performance in convection-dominated problems [3, 4, 5]. We intend to use the SLDG scheme (5.3.3) as a building block for computing exponentials in the commutator-free methods.

Now suppose $u_h^n \in V_h$. Then for $\mathbf{x} \in K$, we have the following algorithm:

Algorithm 5.1. *Commutator-free method.*

```

 $w_h^0 = u_h^n(\mathbf{x})$ 
for  $i = 1 : s$  do
     $U_i = \exp\left(\sum_{j=1}^s \alpha_j^i F_j\right) \cdots \exp\left(\sum_{j=1}^s \alpha_j^1 F_j\right) w^0$ 
     $F_i = \Delta t F_{U_i} = \Delta t \mathbf{V}_i \cdot \nabla$ 
end for
 $u_h^{n+1} = \exp\left(\sum_{j=1}^s \beta_j^s F_j\right) \cdots \exp\left(\sum_{j=1}^s \beta_j^1 F_j\right) w^0$ 

```

where \mathbf{V}_i denotes the value of \mathbf{V} at the intermediate time $t_n^i = t_n + c_i \Delta t$. Meanwhile F_i

represent vector fields frozen at stage values U_i , and $\{\alpha_{ij}^l, \beta_j^l\}$, $i, j = 1, \dots, s$, $l = 1, \dots, J$ are coefficients of the CF method, typically constructed out of s -stage Runge-Kutta methods with coefficients $\{a_{ij}, b_i, c_i\}$, $i, j = 1, \dots, s$ (see [6] for details).

We write ${}^3\hat{F}_{il} = \sum_{j=1}^s \alpha_{ij}^l F_j$, $i = 1, \dots, s+1$, choosing $\alpha_{s+1,j}^l := \beta_j^l$. Then on each element K the flow

$$w_h^i = \exp(\hat{F}_{il}) w_h^0 \quad (5.3.4)$$

would be approximated via the SLDG scheme (5.3.3) as follows

$$\begin{aligned} \frac{1}{\Delta t} \int_K (w_h^i - w_h^0) v \, d\mathbf{x} &= \sum_{j=1}^s \alpha_{ij}^l \int_K [E_{\tau_j}^n w_h^0] \mathbf{V}_j \cdot \nabla v \, d\mathbf{x} \\ &\quad - \alpha_{ij}^l \int_{\partial K} [\widehat{E_{\tau_j}^n w_h^0}] \mathbf{V}_j \cdot \mathbf{n}_K v(s) \, ds, \quad \text{for all } v \in V_h, \end{aligned} \quad (5.3.5)$$

where $\tau_j = c_j \Delta t$. The formula (5.3.5) is adapted from (5.3.3) by replacing the integral over the time interval with the quadrature rule dictated by the CF method.

Generally in the DG as well as the SLDG methods a *flux-limiter* is being introduced at each time step to preserve the monotonicity or positivity of the solution. This has not been exploited in our numerical examples, since the problems considered are linear, and the solution and advecting velocities are fairly smooth. For more discussion on the use of flux-limiters in the context of DG methods we refer to [8, 21].

5.4 Numerical results

5.4.1 Pure advection in 1D

To show the stability of the new SLDG methods over RKDG methods, we consider the constant advection of the Gaussian cone in 1D described by

$$u_t + au_x = 0, \quad x \in \Omega := (-1, 1), \quad t > 0, \quad (5.4.1)$$

where $a = 2$, and the initial data and exact solution are given by the equation

$$u(x, t) = \exp(-(x - x_0 - at)^2 / 2\lambda^2),$$

with $\lambda = \frac{1}{8}$, $x_0 = 0$. We use periodic boundary conditions. The spatial discretization is carried out using the discontinuous Galerkin method with $N_e = 10$ elements and Gauss-Lobatto-Legendre (GLL) quadratures of polynomial degree p . Integration is done up to final time $T = 1$, using n_{steps} time steps corresponding to a Courant number C_r (see [10]). We compare the performance of the SLDG method with RKDG method at various Courant numbers (C_r). For the RKDG method we have used the Godunov-type flux approximation. Both methods are constructed out of the third order RK method described by the Butcher tableau (see e.g. [8, 21])

³For an explicit method, this sum depends on previously computed frozen vector fields F_j , $j < i$.

Chapter 5. On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems

$$\begin{array}{c|ccc} 0 & & & \\ 1 & 1 & & \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & \\ \hline & \frac{1}{6} & \frac{1}{6} & \frac{2}{3} \end{array}.$$

All errors are measured in an element-wise relative L_2 -norm, that is,

$$\|u - u_{exact}\|_{L_2(\Omega)} = \sqrt{\frac{\sum_K \int_K (u - u_{exact})^2}{\sum_K \int_K u_{exact}^2}}$$

The results shown in Table 5.1 clearly reveals that the SLDG methods perform better than the RKDG methods at higher Courant numbers. The RKDG methods become unstable at Courant numbers higher than 1. The SLDG method on the other hand loses some accuracy at lower Courant numbers. This is as a result of accumulation of the SL interpolation error as we take many time steps. A remedy to this is to us higher order interpolation. In Table 5.2 we have again compared the two methods at a fixed Courant number $C_r \approx 0.25$ but different values of polynomial degree p . Both methods exhibit spectral (spatial) order of convergence. Notice that since this Courant number is small, the SLDG only begins to out-perform the RKDG at larger values of p (in this case $p \geq 10$), where SL interpolation errors are minimal.

Also shown in Figure 5.1 are results obtained with CF time integrators (CF122 and CF233) based on second and third order explicit Runge-Kutta methods (respectively) taken from [1], and the Euler method (CF111). The CF122 method is based on the midpoint rule. Integration is done upto a final time $T = 1$, and in each case the number of time steps used is $N_{steps} = 400$ which corresponds to a Courant number $C_r \approx 0.5$. We notice the the CF111 method leads to numerical damping (error = 1.50×10^{-1}), but the second and third order methods CF122 and CF233 yield accurate results (error = 2.00×10^{-3} and 8.38×10^{-6} respectively). The CF111, CF122 and CF233 are constructed out of explicit Runge-Kutta methods with Butcher tableaus given respectively as follows

$$\begin{array}{c|c} 0 & 0 \\ \hline \frac{1}{2} & 1 \end{array}, \quad \begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & \\ \hline & 0 & 1 \end{array}, \quad \begin{array}{c|ccc} 0 & & & \\ \gamma & \gamma & & \\ 1 - \gamma & \gamma - 1 & 2(1 - \gamma) & \\ \hline & 0 & \frac{1}{2} & \frac{1}{2} \end{array},$$

where $\gamma = (3 + \sqrt{3})/6$ (see [1]).

5.4.2 Pure advection in 2D

We now consider the 2D test problem of [7], which involves the advection of a passive tracer in a nondivergent deformational flow. The equation is given by

$$u_t + \mathbf{V} \cdot \nabla u = 0, \quad \text{in } \Omega = (0, 1) \times (0, 1), \quad (5.4.2)$$

Table 5.1: Results for the 1D advection problem for the RKDG and SLDG methods, using various Courant numbers C_r ; $p = 8, N_e = 10, T = 1$.

C_r	n_{steps}	Δt	Relative error (L_2)	
			RKDG	SLDG
0.10	1996	0.0005	1.1699×10^{-6}	2.5513×10^{-5}
0.25	799	0.0013	1.3736×10^{-5}	1.7444×10^{-5}
0.50	400	0.0025	1.0924×10^{-4}	5.9188×10^{-6}
0.75	267	0.0037	3.6679×10^{-4}	4.0203×10^{-6}
1.00	200	0.0050	1.3536×10^{-3}	1.1646×10^{-5}
1.25	160	0.0063	∞	7.3864×10^{-6}
1.49	134	0.0075	∞	3.2610×10^{-4}

Table 5.2: Results for the 1D advection problem for the RKDG and SLDG methods, using various polynomial degrees p ; Courant numbers $C_r \approx 0.25, N_e = 10, T = 1$.

p	C_r	n_{steps}	Δt	Relative error (L_2)	
				RKDG	SLDG
1	0.25	40	0.0250	7.3567×10^{-1}	1.1407×10^0
2	0.25	80	0.0125	2.0851×10^{-1}	4.1497×10^{-1}
4	0.23	250	0.0040	5.1525×10^{-3}	3.8406×10^{-2}
6	0.24	500	0.0020	8.0856×10^{-5}	9.1899×10^{-4}
8	0.25	800	0.0013	1.3685×10^{-5}	1.7460×10^{-5}
10	0.25	1250	0.0008	3.5815×10^{-6}	2.8314×10^{-7}
12	0.24	1750	0.0006	1.3052×10^{-6}	3.6590×10^{-9}

where $u(\mathbf{x}, t)$ represents the scalar field variable and (with $\mathbf{x} := (x_1, x_2)$) the velocity field is given as

$$\mathbf{V}(\mathbf{x}, t) = \begin{bmatrix} \sin^2(\pi x_1) \sin(2\pi x_2) \cos(\pi t/5) \\ -\sin^2(\pi x_2) \sin(2\pi x_1) \cos(\pi t/5) \end{bmatrix}.$$

The boundary condition is homogeneous Dirichlet and the initial data is

$$u(\mathbf{x}, 0) = \frac{1 + \cos(\pi r)}{2}, \text{ with } r := r(\mathbf{x}) = \min \left(1, 4 \sqrt{\left(x_1 - \frac{1}{4}\right)^2 + \left(x_2 - \frac{1}{4}\right)^2} \right).$$

We solve this problem using the CF233 method. The spatial domain is sub-divided into $N_e = 10 \times 10 = 100$ elements. The fourth order explicit Runge-Kutta method is used

Chapter 5. On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems

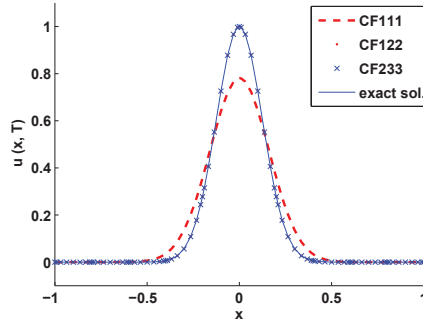


Figure 5.1: SLDG methods with $p = 8$, $N_e = 10$ for Gauss cone advection in 1D, using first, second and third order time integrators CF111, CF122 and CF233. The exact solution is represented by the solid curve. The respective errors in the relative L_2 norm are 1.50×10^{-1} , 2.00×10^{-3} , 8.38×10^{-6} . Meanwhile $T = 1$, $n_{steps} = 400$ and $C_r \approx 0.5$

to solve accurately the characteristic equation (5.3.2). The initial cone is swirled around in the anti-clockwise direction, being deformed in the process, until it comes to a stationary position at time $t = 2.5$. It is then driven in the reverse direction so that at time $t = 5$ the initial cone is being re-formed. Therefore the exact solution at time $t = 5$ coincides with the initial data. The results are shown in Figure 5.2.

We start by using $N_e = 10 \times 10$ elements with polynomial degree $p = 10$ in each element (see first column in Figure 5.2). The time step chosen here is $\Delta t = 1/120$ which corresponds to a Courant number of $C_r = 2.5253$. The results show some overshoots and undershoots in the solution. Also the exact solution is not fully recovered. The L_2 -error obtained in the case is 0.0063. Next we use a refined mesh, by choosing $p = 15$ and $N_e = 10 \times 10$. The time step used here is $\Delta t = 1/240$ which corresponds to a Courant number of $C_r = 2.7370$. The solution is greatly improved (see second column in Figure 5.2). We obtain an error of 0.0019 in this case. We also observed that at these choices of Courant numbers the RKDG schemes were unstable.

Conclusion

We have established a relation between the SLDG methods and the commutator-free (CF) methods for solving linear advection problems with divergence-free velocities. We have also shown (via numerical experiments) that the methods have better stability at high Courant numbers than the RKDG methods, and maintain the good spatial accuracy of DG methods. The model problems considered contain advection velocities that are constant or dependent on both time and space. The main advantage of the CF methods is that they provide routines for accurate approximations of linear pure advection. We believe that

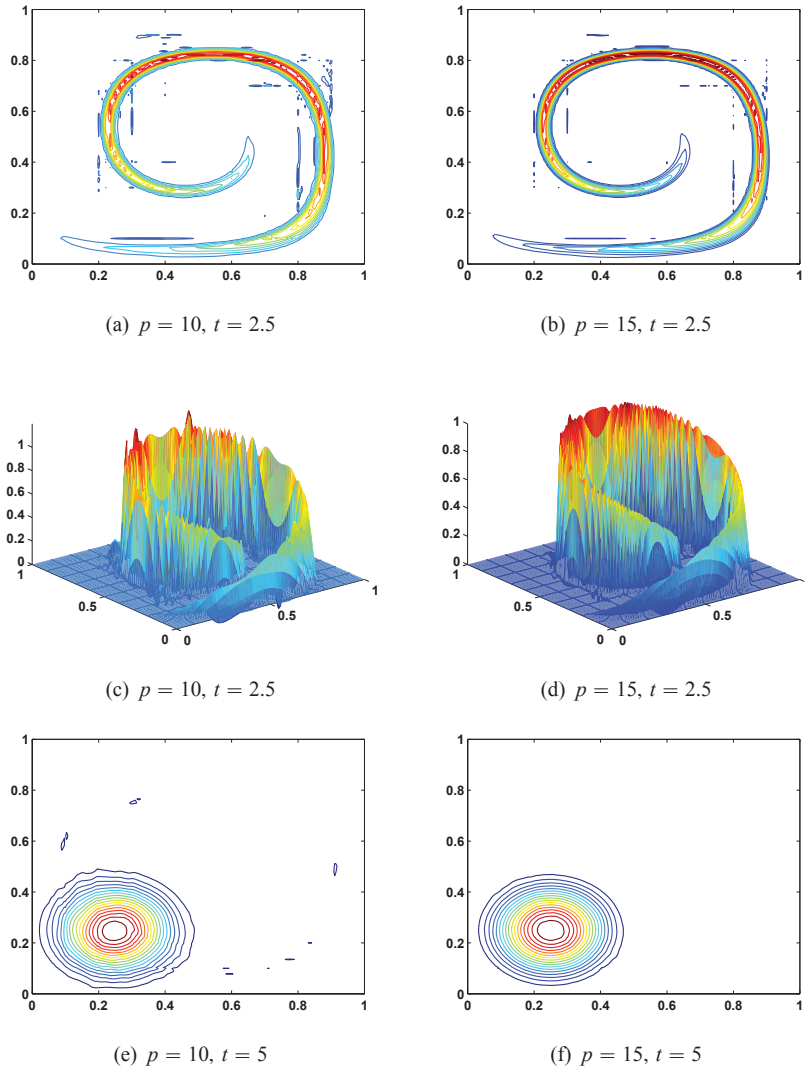


Figure 5.2: Tracer advection in 2D using CF233. In the first column, $p = 10$. In the second column, $p = 15$. The number of elements used in both case is $N_e = 10 \times 10$. First row: solution contours at $t = 2.5$; Second row: solutions at $t = 2.5$; third row: solution contours at $t = 5$. Contour levels: -36 to 36 by 18 .

these preliminary results are promising and that it is worth pursuing the study of the SLDG combined with CF methods in the case where the advection term is nonlinear. This idea,

Chapter 5. On discontinuous Galerkin methods and commutator-free exponential integrators for advection problems

however, has not been investigated here. Applications to nonlinear convection-diffusion models would also be of interest. A recent study on this is the work of Qiu and Shu in [17]

Acknowledgements

I am especially grateful to my PhD supervisor Elena Celledoni for her guidance on the application of the commutator-free Lie group exponential integrators. My sincere gratitude also goes to Franco Brezzi, Donatella Marini and Edré Süli for the lectures at the 2010 summer school on Discontinuous Galerkin Methods, Dobbiacco, Italy.

References

- [1] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri, *Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations*, Appl. Numer. Math. **25** (1997), 151–167.
- [2] C. E. Baumann and J. T. Oden, *A discontinuous hp finite element method for convection-diffusion problems*, Comput. Methods Appl. Mech. Engrg. **175** (1999), no. 3-4, 311–341. MR 1702201 (2000d:65171)
- [3] E. Celledoni, *Eulerian and semi-Lagrangian commutator-free exponential integrators*, CRM Proceedings and Lecture Notes **39** (2005), 77–90.
- [4] E. Celledoni, *Semi-Lagrangian methods and new integrators for convection dominated problems*, Oberwolfach Reports **12** (2006).
- [5] E. Celledoni and B. K. Kometa, *Semi-Lagrangian Runge-Kutta exponential integrators for convection dominated problems*, J. Sci. Comput. **41** (2009), no. 1, 139–164.
- [6] E. Celledoni, A. Marthinsen, and B. Owren, *Commutator-free Lie group methods*, FGCS **19** (2003).
- [7] A. Chertock and A. Kurganov, *On a practical implementation of particle methods*, Appl. Numer. Math. **56** (2006), no. 10-11, 1418–1431. MR 2245465 (2007k:65014)
- [8] B. Cockburn and C.-W. Shu, *Runge-Kutta discontinuous Galerkin methods for convection-dominated problems*, J. Sci. Comput. **16** (2001), no. 3, 173–261.
- [9] R. E. Ewing and H. Wang, *A summary of numerical methods for time-dependent advection-dominated partial differential equations*, J. Comput. Appl. Math. **128** (2001), no. 1-2, 423–445, Numerical analysis 2000, Vol. VII, Partial differential equations. MR 1820884 (2002a:65003)

-
- [10] F. X. Giraldo, *The Lagrange-Galerkin spectral element method on unstructured quadrilateral grids*, J. Comput. Phys. **147** (1998), no. 1, 114–146. MR 1657765 (99j:65177)
- [11] S. Gottlieb, D. I. Ketcheson, and C.-W. Shu, *High order strong stability preserving time discretizations*, J. Sci. Comput. **38** (2009), no. 3, 251–289.
- [12] S. Gottlieb, C.-W. Shu, and E. Tadmor, *Strong stability-preserving high-order time discretization methods*, SIAM Rev. **43** (2001), no. 1, 89–112 (electronic).
- [13] P. Houston, C. Schwab, and E. Süli, *Discontinuous hp-finite element methods for advection-diffusion-reaction problems*, SIAM J. Numer. Anal. **39** (2002), no. 6, 2133–2163 (electronic). MR 1897953 (2003d:65108)
- [14] P. Lasaint and P.-A. Raviart, *On a finite element method for solving the neutron transport equation*, Mathematical aspects of finite elements in partial differential equations (Proc. Sympos., Math. Res. Center, Univ. Wisconsin, Madison, Wis., 1974), Math. Res. Center, Univ. of Wisconsin-Madison, Academic Press, New York, 1974, pp. 89–123. Publication No. 33.
- [15] K. W. Morton, *On the analysis of finite volume methods for evolutionary problems*, SIAM J. Numer. Anal. **35** (1998), no. 6, 2195–2222. MR 1655843 (99m:65172)
- [16] J. T. Oden, I. Babuška, and C. E. Baumann, *A discontinuous hp finite element method for diffusion problems*, J. Comput. Phys. **146** (1998), no. 2, 491–519. MR 1654911 (99m:65173)
- [17] J.-M. Qiu and C.-W. Shu, *Positivity preserving and semi-Lagrangian discontinuous Galerkin formulation: theoretical analysis and application to the Vlasov-Poisson system*, Technical report, 2011-1, Scientific Computing Group, Division of Applied Mathematics, Brown University, <http://www.dam.brown.edu/scicomp/>, 2011.
- [18] W. H. Reed and T. R. Hill, *Triangular mesh methods for the neutron transport equation*, Los Alamos Scientific Laboratory Report LA-UR-73-479, 1973.
- [19] M. Restelli, L. Bonaventura, and R. Sacco, *A semi-Lagrangian discontinuous Galerkin method for scalar advection by incompressible flows*, J. Comput. Phys. **216** (2006), no. 1, 195–215.
- [20] C.-W. Shu, *Total-variation-diminishing time discretizations*, SIAM J. Sci. Statist. Comput. **9** (1988), no. 6, 1073–1084.
- [21] ———, *Discontinuous Galerkin methods: General approach and stability*, Lecture Notes, Division of Applied Mathematics, Brown University, Providence, RI 02912, USA, 2008.

Paper V

Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

Elena Celledoni, Bawfeh Kingsley Kometa and Olivier
Verdier

*Preprint series: Numerics, 2011-07, Department of
Mathematics, NTNU, Norway*

Chapter 6

Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

Abstract. Direct applications of high order DIRK-CF methods as presented in [7] to the incompressible Navier-Stokes equations were found to yield a loss in order of convergence. The DIRK-CF methods are exponential integrators arising from the IMEX Runge-Kutta techniques proposed in [1], and are semi-Lagrangian when applied to convection diffusion equations. As discussed in [17], inappropriate implementation of projection methods for incompressible flows can lead to a loss in the order of convergence. In this paper we recover the full order of the IMEX methods using projections onto the space of divergence-free vector fields and we discuss the difficulties encountered in using similar techniques for the semi-Lagrangian DIRK-CF methods. We finally assess the performance of the semi-Lagrangian DIRK-CF methods for the Navier-Stokes equations in convection-dominated problems.

6.1 Introduction

Consider the incompressible Navier-Stokes equations

$$\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u} = \nu \nabla^2 \mathbf{u} - \nabla p \quad (6.1.1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (6.1.2)$$

$$\mathbf{u}|_{\partial\Omega} = 0, \quad (6.1.3)$$

here $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ on the cylinder $\Omega \times [0, T]$ is the velocity field ($\Omega \subset \mathbf{R}^d$ and $d = 2, 3$), subjected to the incompressibility constraint (6.1.2), $p = p(\mathbf{x}, t)$ is the pressure and plays

Chapter 6. Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

the role of a Lagrange multiplier, and ν is the kinematic viscosity of the fluid. We consider no slip or periodic boundary conditions

$$\mathbf{u}|_{\partial\Omega} = \mathbf{u}_b, \quad (6.1.4)$$

$$\mathbf{u} \quad \text{periodic.} \quad (6.1.5)$$

In the case of no slip boundary conditions we will also use that $\mathbf{u}_b \cdot \mathbf{n} = 0$ where \mathbf{n} is the unit normal to the boundary $\partial\Omega$. For no slip boundary conditions we will mostly consider the case

$$\mathbf{u}_b = 0. \quad (6.1.6)$$

The variables (\mathbf{u}, p) are sometimes called primitive variables and the accurate approximation of both these variables is desirable in numerical simulations.

In this paper we study semi-Lagrangian discretization methods in time to be used in combination with high order spatial discretizations of the Navier-Stokes equations, like for example spectral element methods. High order methods are particularly interesting in cases when highly accurate numerical approximations of a given flow case are required. A relevant situation is the direct numerical simulation of turbulence phenomena (DNS), as pointed out for example in [23].

The methods we consider here are implicit-explicit methods of Runge-Kutta type which we named DIRK-CF, and they have been proposed in [6, 7]. These methods arise from IMEX techniques proposed in [2, 1]. In addition to being implicit-explicit the methods are semi-Lagrangian and they show improved performance in convection-dominated problems. So far the case of linear and nonlinear convection diffusion equations have been considered.

It is our goal in this paper to further investigate the extension of these methods to the incompressible Navier-Stokes equations and to assess their performance. Given a time-stepping technique, a very used approach to adapt the method to the incompressible Navier-Stokes equations is by means of projections. The primary example of this technique, and most famous projection method for the incompressible Navier-Stokes equations is the Chorin's projection method, proposed by Chorin in [9, 10] and Témam [22]. Chorin's method is a version of the implicit Euler integration method adapted to the Navier-Stokes equations.

The study of the temporal order of this method was considered in [20, 21] and it revealed order one for the velocity and only $\frac{1}{2}$ for the approximation of the pressure. This and similar order reduction phenomena are typical of projection methods for Navier-Stokes equations and must be handled properly to achieve higher order. Lately a better understanding of the issues of order reduction in a variety of projection methods, and remedies to this problem appeared in [19, 3, 17].

We consider projection methods for IMEX Runge-Kutta schemes as a starting point to discuss the extension of the methods of [7] to the Navier-Stokes equations. In this preliminary work we explain some of the difficulties encountered in the case of the semi-Lagrangian methods, spectral element space discretizations and the Navier-Stokes

equations. We obtain methods of IMEX type which show up to third order temporal accuracy in the velocity and first order in the pressure. The semi-Lagrangian methods achieve up to second temporal order in the velocity.

In section 6.2 we consider appropriate projections to be used in the reformulation of our methods in the context of Navier-Stokes equations, including some relevant background material. In section 6.3 we discuss implicit-explicit methods, and the semi-Lagrangian methods named DIRK-CF and their extensions to Navier-Stokes equations. Section 6.4 is devoted to numerical experiments. In this section we provide numerical verification of the temporal order of the methods; we illustrate the benefits of the proposed semi-Lagrangian methods in the case of convection-dominated problems; we also devote this section to the description of the implementation details behind our numerical results.

6.2 Projection methods for the incompressible Navier-Stokes equations

6.2.1 Leray projector

According to the Helmholtz decomposition of vector fields, $\mathbf{w} \in (L^2(\mathbf{R}^d))^d$ can be decomposed into a curl-free and a divergence-free part:

$$\mathbf{w} = \nabla\phi + \mathbf{v}, \quad \nabla \cdot \mathbf{v} = 0. \quad (6.2.1)$$

We are interested in such decomposition on bounded domains Ω , taking into account boundary conditions. We consider a projection on the subset of the space of divergence free vector fields, with prescribed boundary conditions on $\partial\Omega$:

$$H = \{\mathbf{v} \in (L^2(\Omega, \mathbf{R}^d))^d \mid \nabla \cdot \mathbf{v} = 0, \mathbf{v}|_{\partial\Omega} = \text{bc}\}, \quad \mathcal{P} : \mathcal{W} \subset (L^2(\Omega, \mathbf{R}^d))^d \rightarrow H,$$

and $\mathcal{W} \subset (L^2(\Omega, \mathbf{R}^d))^d$ an appropriate subset of $(L^2(\Omega, \mathbf{R}^d))^d$, here the boundary conditions (bc) are either periodic or $\mathbf{n} \cdot \mathbf{v}|_{\partial\Omega} = 0$.

So \mathcal{P} is such that

$$\mathcal{P}(\mathbf{w}) = \mathbf{v}, \quad (6.2.2)$$

satisfying the conditions

$$\nabla \cdot \mathbf{v} = 0, \quad \mathbf{v}|_{\partial\Omega} = \text{bc}. \quad (6.2.3)$$

Assuming \mathbf{w} satisfies boundary conditions compatible with \mathbf{v} (say \mathbf{w} periodic or with no slip boundary conditions), we can take \mathcal{P} to be the Leray projector [14]. This projector is constructed by taking \mathbf{v} as

$$\mathcal{P}(\mathbf{w}) = \mathbf{v} = \mathbf{w} - \nabla\phi,$$

where ϕ is the solution of the Poisson equation

$$\nabla^2\phi = \nabla \cdot \mathbf{w} \quad (6.2.4)$$

and boundary conditions for ϕ either periodic or Neumann:

$$0 = \mathbf{n} \cdot \mathbf{v}|_{\partial\Omega} = \mathbf{n} \cdot \mathbf{w}|_{\partial\Omega} - \mathbf{n} \cdot \nabla\phi|_{\partial\Omega}. \quad (6.2.5)$$

Chapter 6. Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

6.2.2 Incompressible Navier-Stokes and projections

In general, taking the divergence of the momentum equation, (6.1.1), we obtain a Poisson equation for the pressure

$$\nabla^2 p = \nabla \cdot (\nu \Delta \mathbf{u} - \mathbf{u} \cdot \nabla \mathbf{u}). \quad (6.2.6)$$

When \mathbf{u} is space-periodic, i.e. (6.1.5), the pressure p is fully defined in terms of the velocity field \mathbf{u} and the periodicity condition. In the case of no slip boundary conditions, (6.1.4) and (6.1.6), solving the Poisson equation for p by imposing

$$\frac{\partial p}{\partial \mathbf{n}} = \nu \Delta \mathbf{u} \cdot \mathbf{n},$$

on the boundary, fully determines the pressure. In both cases we can write $p = \psi(\mathbf{u})$, [14]. We can then eliminate the pressure from the momentum equation and obtain

$$\mathbf{u}_t - \nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{u}_t - \nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla \psi(\mathbf{u}) = 0.$$

We observe that for \mathbf{u} satisfying the Navier-Stokes equations (6.1.1-6.1.3) we have

$$\mathcal{P}(\mathbf{u}) = \mathbf{u}, \quad \mathcal{P}(\mathbf{u}_t) = \mathbf{u}_t, \quad \mathcal{P}(\nabla p) = 0,$$

and we can rewrite the Navier-Stokes equations as

$$\mathbf{u}_t = \mathcal{P}(\nu \Delta \mathbf{u} - \mathbf{u} \cdot \nabla \mathbf{u} - \nabla p). \quad (6.2.7)$$

An alternative formulation [14] is

$$\mathbf{u}_t = \nu \mathcal{P}(\Delta \mathbf{u}) - \mathcal{P}(\mathbf{u} \cdot \nabla \mathbf{u}), \quad (6.2.8)$$

where the two projections correspond to two different Poisson problems which have both periodic or Neumann boundary conditions to be imposed on corresponding Lagrangian multipliers.

In the context of IMEX and semi-Lagrangian Runge-Kutta time integration methods, the formulation (6.2.8) seems to be the most appropriate. The intention is to apply different Runge-Kutta coefficients to the convection operator and the diffusion operator. However non-trivial complications arise when discretizing in space.

After spatial discretizations of type spectral-Galerkin or spectral element methods, we obtain a system of differential-algebraic equations of the type:

$$B\dot{y} = Ay + C(y)y - D^T z, \quad Dy = 0, \quad (6.2.9)$$

which should be satisfied with appropriate boundary conditions. Here A is the discrete Laplacian, B is the mass matrix, $C(y)$ is the discrete convection operator, D is the discrete divergence and D^T is the discrete gradient. The numerical solution $y \approx \mathbf{u}$ includes values pertaining to boundary nodes, and the discrete operators are sized accordingly. The intention is to impose the boundary conditions directly on the numerical approximation

6.2. Projection methods for the incompressible Navier-Stokes equations

y . Boundary conditions are not inbuilt in (6.2.9) as in the case of finite differences discretizations, and are enforced by applying an operator R_b to the numerical solution.

If D is full rank, the Lagrangian multiplier z in (6.2.9) can be obtained by solving the linear system

$$DB^{-1}D^T z = DAy + DC(y)y, \quad (6.2.10)$$

but such z is not necessarily satisfying the boundary conditions satisfied by the pressure in (6.2.6) deduced from (6.1.1), and similarly \dot{y} is not satisfying the boundary conditions satisfied by \mathbf{u}_t in (6.1.1). Assuming Π denotes the projection on the space of discrete divergence free vector fields, regardless of boundary conditions, this gives

$$\Pi B^{-1}(Ay + C(y)y) = B^{-1}(Ay + C(y)y - D^T z) \quad (6.2.11)$$

and z the solution of (6.2.10), we can introduce the discrete analogs to (6.2.7) and (6.2.8) simply as

$$\dot{y} = \Pi B^{-1}(Ay + C(y)y),$$

and

$$\dot{y} = \Pi B^{-1}(Ay) + \Pi(C(y)y).$$

Applying Runge-Kutta methods, IMEX methods or semi-Lagrangian exponential integrators to these equations will produce approximations of \mathbf{u} which are divergence free, but do not, in general, satisfy the desired boundary conditions. Trying to enforce boundary conditions by using instead projections $\tilde{\Pi}$ mapping Ay and $C(y)y$ into the space of divergence free vector fields with appropriate boundary conditions, turns out to be ill-conditioned. Another inconvenience coming from the type of discretizations considered in this work, is that the pressure is not defined on boundary nodes and the boundary conditions cannot be imposed on the pressure (as assumed for the solution of the Poisson equations pertaining to \mathcal{P}). The only acceptable alternative is to impose boundary conditions directly on the numerical approximations of the solution, i.e. the stage values of the Runge-Kutta method. We then obtain that the boundary conditions satisfied by \mathbf{u}_t are respected at the discrete level only for some appropriate, numerical, discrete derivatives.

In the next section we will show how this is handled successfully in the case of IMEX methods.

The relation between (6.1.1), (6.2.8) and (6.2.7) in terms of the corresponding Lagrangian multipliers might be important in order to obtain accurate approximations of the pressure. For example in the periodic case we get

$$\mathcal{P}(\Delta \mathbf{u}) = \Delta \mathbf{u} \Rightarrow \mathbf{u} \cdot \nabla \mathbf{u} + \nabla \psi(\mathbf{u}) = \mathcal{P}(\mathbf{u} \cdot \nabla \mathbf{u})$$

but in the no-slip case this is not so.

We however always have

$$\mathbf{u} \cdot \nabla \mathbf{u} + \nabla \psi(\mathbf{u}) = (I - \mathcal{P})(\nu \Delta \mathbf{u}) + \mathcal{P}(\mathbf{u} \cdot \nabla \mathbf{u}),$$

here I denotes the identity operator.

6.3 High order implicit-explicit and semi-Lagrangian methods of Runge-Kutta type

6.3.1 IMEX Runge-Kutta

We consider IMEX methods with a DIRK (diagonally implicit Runge-Kutta) implicit part to be applied to the diffusion operator and an appropriate explicit part to be used for the convection operator. Applied to (6.1.1) the projected IMEX methods are

$$\mathbf{U}_i = \mathcal{P}(\mathbf{u}_n + \Delta t \sum_{j=1}^{i-1} (a_{i,j}(\Delta \mathbf{U}_j - \nabla P_j) - \tilde{a}_{i,j} \mathbf{U}_j \cdot \nabla \mathbf{U}_j) + \Delta t a_{i,i} \Delta \mathbf{U}_i), \quad i = 1, \dots, s$$

and P_i is the Lagrangian multiplier to be used to perform the projection \mathcal{P} . We assume both the Runge-Kutta methods with coefficients $\{a_{i,j}\}_{i,j=1,\dots,s}$ and $\{\tilde{a}_{i,j}\}_{i,j=1,\dots,s}$ respectively, are stiffly accurate, so, $\mathbf{u}_{n+1} = \mathbf{U}_s$. To obtain the fully discrete version of the methods we apply them first to the equation (6.2.9) and obtain:

$$BY_i = By_n + \Delta t \sum_{j=1}^{i-1} [a_{i,j}(AY_j - D^T Z_j) - \tilde{a}_{i,j} C(Y_j) Y_j] + \Delta t a_{i,i}(AY_i - D^T Z_i), \quad i = 1, \dots, s$$

with the constraint $DY_i = 0$. We next apply the operator R_b enforcing boundary conditions on Y_i , and finally we solve the following linear system for Y_i and Z_i ,

$$\begin{aligned} R_b(B - \Delta t a_{i,i} A) Y_i + \Delta t a_{i,i} R_b D^T Z_i &= R_b(By_n + \Delta t \sum_{j=1}^{i-1} [a_{i,j}(AY_j - D^T Z_j) + \tilde{a}_{i,j} C(Y_j) Y_j]) \\ DY_i &= 0. \end{aligned}$$

The solution of such linear system is obtained by a Schur-complement approach and the inversion of the discrete Helmholtz operator

$$R_b(B - \Delta t a_{i,i} A)$$

by applying a preconditioned conjugate gradient algorithm. We obtain that $y_{n+1} = Y_s$ is the approximation of the velocity field at time t_{n+1} and Z_s is the corresponding approximation of the pressure.

6.3.2 Semi-Lagrangian IMEX Runge-Kutta

We here consider a second order method presented in [7] in the case of convection diffusion equations. We refer to [7] for the general formulation of these methods, which are named DIRK-CF. We apply the method to (6.2.9), the first stage is

$$Y_1 = y_n, \quad Z_1 = 0, \quad \varphi_1 = I.$$

Defining $\varphi_2 = \exp(\Delta t \tilde{a}_{2,1} C(Y_1))$ the second stage is

$$Y_2 = \varphi_2 [y_n + \Delta t a_{2,1} \varphi_1^{-1} B^{-1} (AY_1 - D^T Z_1)] + \Delta t a_{2,2} B^{-1} (AY_2 - D^T Z_2),$$

with $DY_2 = 0$. The term $D^T Z_1 = 0$. We now multiply both sides by B and apply R_b to obtain a linear system for Y_2 and Z_2 . This linear system is

$$\begin{aligned} R_b(B - \Delta t a_{2,2} A) Y_2 + \Delta t a_{2,2} R_b D^T Z_2 &= R_b B \varphi_2 (y_n + \Delta t a_{2,1} B^{-1} AY_1), \\ DY_2 &= 0. \end{aligned}$$

We interpret the $\varphi_1 w$ as the transport of w along the flow of the vector field Y_1 .

At the third stage, we define $\varphi_3 = \exp(\Delta t \tilde{a}_{3,1} C(Y_1) + \Delta t \tilde{a}_{3,2} C(Y_2))$ and write

$$\begin{aligned} Y_3 = \varphi_3 [y_n + \Delta t a_{3,1} \varphi_1^{-1} B^{-1} (AY_1 - D^T Z_1) + \Delta t a_{3,2} \varphi_2^{-1} B^{-1} (AY_2 - D^T Z_2)] \\ + \Delta t a_{3,3} B^{-1} (AY_3 - D^T Z_3), \end{aligned}$$

with $DY_3 = 0$. After applying R_b we obtain the linear system

$$\begin{aligned} R_b(B - \Delta t a_{3,3} A) Y_3 + \Delta t a_{3,3} R_b D^T Z_3 &= R_b B \varphi_3 (y_n + \Delta t a_{3,1} B^{-1} AY_1 \\ &\quad + \Delta t a_{3,2} \varphi_2^{-1} B^{-1} (AY_2 - D^T Z_2)), \\ DY_3 &= 0. \end{aligned}$$

We finally take $y_{n+1} = Y_3$. This approach to enforce boundary conditions for the DIRK-CF methods is the straightforward counterpart of the approach used for IMEX methods in the previous section, and leads to methods with temporal order at most 2 in the velocity. We were unable to obtain order three or more with this technique.

6.4 Numerical experiments

For the numerical experiments we shall employ a spectral element method (SEM) based on the standard Galerkin weak formulation as detailed out in [13]. We use a rectangular domain consisting of N_e uniform elements. The approximation is done in $\mathbb{P}_N - \mathbb{P}_{N-2}$ compatible velocity-pressure discrete spaces, i.e., keeping the time variable t fixed, in each element we approximate the velocity by a N -degree Lagrange polynomial based on Gauss-Lobatto-Legendre (GLL) nodes in each spatial coordinate, and the pressure by $(N - 2)$ -degree Lagrange polynomial based on Gauss-Legendre (GL) nodes. The discrete spaces are spanned by tensor product polynomial basis functions. The resulting discrete system has the form (6.2.9). We begin by describing some key implementation issues involved in the numerical experiments.

Chapter 6. Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

6.4.1 Implementation issues

Pressure-splitting scheme

This scheme is used to enhance solving the linear Stokes systems [12] occurring at each stage of an IMEX or DIRK-CF method. Suppose

$$\begin{aligned} \frac{1}{\gamma\Delta t}BY_i - AY_i - D^T Z_i &= B\hat{y}_n \\ DY_i &= 0 \end{aligned} \quad (6.4.1)$$

represents a linear Stokes system arising from stage i of a first or second order IMEX or DIRK-CF method applied to (6.2.9), where γ is a parameter of the method. Here the variable \hat{y}_n incorporates the explicit treatment of the convection, the initial data and vector fields at earlier stages. The splitting scheme (irrespective of boundary conditions) is done in the following steps:

$$\text{Step 1: } \frac{1}{\gamma\Delta t}B\hat{Y}_i - A\hat{Y}_i - D^T p_n = B\hat{y}_n$$

$$\text{Step 2: } D^T B^{-1}D\delta p_i = -\frac{1}{\gamma\Delta t}D\hat{Y}_i$$

$$\text{Step 3: } Y_i = \hat{Y}_i - \gamma\Delta t B^{-1}D^T \delta p_i, \quad Z_i = p_n + \delta p_i.$$

Step 1 is an explicit approximation of the stage value of the velocity using the initial pressure. This approximation is not divergence-free. Step 2 and 3 are thus the projection steps which enforce the divergence-free constrain and correct the velocity and pressure. Note that this approximation introduces a truncation error of order 3, and is thus sufficient for methods order up to 2 (see e.g.[12]). Solving (6.4.1) directly would lead to solving equations with the operator $D^T H D$ (with $H := \frac{1}{\gamma\Delta t}B + A$) for the pressure. However, the cost of inverting $D^T H^{-1}D$ is much higher than for inverting $D^T B^{-1}D$ in Step 2, since B is usually diagonal or tridiagonal an easier to invert than H . This explains the main advantage for using the pressure-splitting schemes in the numerical computations. We have exploited this advantage in the numerical experiments in sections 6.4.4 and 6.4.5.

Boundary conditions

We illustrate the strategy for implementing the boundary conditions in the context of spectral element methods. Let R_p represent a periodic boundary operator, defined such that for a given vector y in the solution space or space of vector fields, $R_p y$ is periodic. Each stage of an IMEX or DIRK-CF method applied to (6.2.9) can be expressed in the form (6.4.1). Multiplying the first equation of (6.4.1) by R_p we obtain the system

$$\begin{aligned} HY_i - R_p D^T Z_i &= R_p B\hat{y}_n \\ DY_i &= 0 \end{aligned} \quad (6.4.2)$$

where $H := R_p(\frac{1}{\gamma\Delta t}B - A)$. The matrix H results from the discrete Helmholtz operator and is symmetric positive-definite (SPD); the mass B is diagonal and SPD, and thus easy to invert. The entire system (6.4.2) forms a symmetric saddle system, which has a unique solution for Y_i provided D is of full rank. The choice of spatial discretization method provides a full-rank matrix D . The system (6.4.2) can be solved by a Schur-complement approach and the pressure-splitting scheme.

The treatment of Dirichlet boundary conditions is very similar and we refer to [12] for further details. In the experiments reported in this paper, no special treatment has been taken to enforce pressure boundary conditions, since the discrete pressure space is not explicitly defined on discretization nodes on the boundary.

6.4.2 Temporal order tests for the IMEX methods

We investigate numerically the temporal order of convergence of some IMEX-RK methods as described in section 6.3.1. The methods considered here are the second and third order IMEX-RK schemes with stiffly-accurate and L-stable DIRK parts [1]. We refer to them as IMEX2L and IMEX3L respectively. They are given by the Butcher tableaus in Tables 6.1 and 6.2 where $\gamma = (2 - \sqrt{2})/2$ and $\delta = 1 - 1/(2\gamma)$.

In the first example we consider the Taylor vortex problem with exact solution and initial data given by

Table 6.1: IMEX2L: $\gamma = (2 - \sqrt{2})/2$ and $\delta = 1 - 1/(2\gamma)$

0	
γ	γ
1	$1 - \gamma \quad \gamma$
	$1 - \gamma \quad \gamma$

0		
γ	γ	
1	$\delta \quad 1 - \delta$	
	$\delta \quad 1 - \delta \quad 0$	

Table 6.2: IMEX3L

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

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{2}{3}$	$\frac{11}{18}$	$\frac{1}{18}$		
$\frac{1}{2}$	$\frac{5}{6}$	$-\frac{5}{6}$	$\frac{1}{2}$	
1	$\frac{1}{4}$	$\frac{7}{4}$	$\frac{3}{4}$	$-\frac{7}{4}$
	$\frac{1}{4}$	$\frac{7}{4}$	$\frac{3}{4}$	$-\frac{7}{4} \quad 0$

Chapter 6. Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

$$\begin{cases} u_1 = -\cos(\pi x_1) \sin(\pi x_2) \exp(-2\pi^2 t/Re), \\ u_2 = \sin(\pi x_1) \cos(\pi x_2) \exp(-2\pi^2 t/Re), \\ p = -\frac{1}{4}[\cos(2\pi x_1) + \cos(2\pi x_2)] \exp(-4\pi^2 t/Re), \end{cases} \quad (6.4.3)$$

writing $Re = 1/\nu$ for the Reynolds number, and $\mathbf{u} := (u_1, u_2)$, $\mathbf{x} := (x_1, x_2)$. The boundary condition is doubly-periodic on the domain $x_1, x_2 \in [-1, 1]$, and we choose $Re = 2\pi^2$. For the spatial discretization we use a spectral method of order $N = 12$, and the time integration is done up to time $T = 1$. For each step size $\Delta t = T/2^k$, $k = 1, \dots, 6$, the error between the numerical solution and the exact PDE solution (at time T) are measured in the L_2 -norm, for both the velocity and pressure. The results for both the IMEX2L and IMEX3L show a temporal convergence of order 2 and 3 respectively (see Figure 6.1).

Similar experiments are carried out for the test problem [18] with exact solution given by

$$\begin{cases} u_1 = \pi \sin(2\pi x_2) \sin^2(\pi x_1) \sin t, \\ u_2 = -\pi \sin(2\pi x_1) \sin^2(\pi x_2) \sin t, \\ p = \cos(\pi x_1) \sin(\pi x_2) \sin t, \end{cases} \quad (6.4.4)$$

for $x_1, x_2 \in [0, 1]$ and $t \in [0, T]$, with $T = 1$. A corresponding forcing term is added to (6.1.1) for a given Reynolds number. In this test case we have used $Re = 100$. This time the boundary condition used is homogeneous Dirichlet. The results are shown in Figure 6.2.

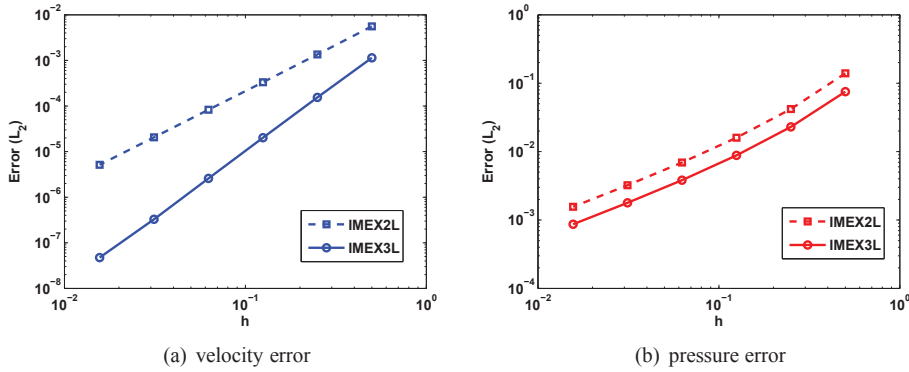


Figure 6.1: Order of convergence of the IMEX2L and IMEX3L. Test problem: Taylor vortex (6.4.3); $Re = 2\pi^2$, $T = 1$, $N = 12$, $N_e = 1$, $\Omega = [-1, 1]^2$, $h = \Delta t = T/2^k$, $k = 1, \dots, 6$. bc: periodic. **(a)** velocity error: IMEX2L (slope = 2.0154), IMEX3L (slope = 2.9250); **(b)** pressure error: IMEX2L (slope = 1.2773), IMEX3L (slope = 1.2711).

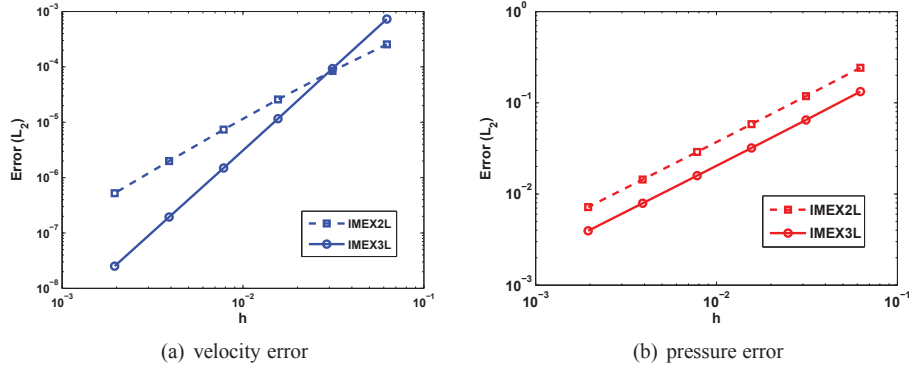


Figure 6.2: Order of convergence of the IMEX2L and IMEX3L. Test problem:(6.4.4); $Re = 100$, $T = 1$, $N = 16$, $N_e = 1$, $\Omega = [0, 1]^2$, $h = \Delta t = T/2^k$, $k = 4, \dots, 9$. bc: homogeneous Dirichlet. **(a)** velocity error: IMEX2L (slope = 1.7908), IMEX3L (slope = 2.9669); **(b)** pressure error: IMEX2L (slope = 1.0140), IMEX3L (slope = 1.0132).

6.4.3 Temporal order tests for the DIRK-CF methods

Using the IMEX2L and IMEX3L methods, we construct two DIRK-CF methods, namely, DIRK-CF2L and DIRK-CF3L, of classical orders 2 and 3 respectively. DIRK-CF2L is applied to (6.2.9) following the algorithm discussed in section 6.3.2. For DIRK-CF3L we use a similar algorithm at each stage, but an extra update stage added, followed by a projection step to enforce the divergence-free condition. We obtain second order for DIRK-CF2L, but DIRK-CF3L suffer a loss in order (see Figure 6.3). The flows of the convecting vector fields are computed in a semi-Lagrangian fashion. We believe that the implementation of the boundary conditions alongside the projections is still not very clear from a numerical point of view. The test problem used is the Taylor vortex problem (6.4.3) with doubly-periodic domain $x_1, x_2 \in [-1, 1]$, and we choose $Re = 2\pi^2$. For the spatial discretization we use a spectral method of order $N = 12$, and the time integration is done up to time $T = 1$. For each step size $\Delta t = T/2^k$, $k = 4, \dots, 9$, the velocity error between the numerical solution and the exact PDE solution (at time T) is measured in the L_2 -norm. Meanwhile the pressure error shows first order of convergence (see Figure 6.3b).

The numerical experiments presented in sections 6.4.4 and 6.4.5 illustrate the potentials of the semi-Lagrangian exponential integrators [7] for the treatment of convection-dominated problems. We consider two examples involving high Reynolds incompressible Navier-Stokes models. These examples are the shear layer roll-up problem [4, 11, 13], and the 2D lid-driven cavity problem (see [16, 5] and references therein). The second order semi-Lagrangian DIRK-CF2L method (named SL2L in [7]) is used in each of these experiments. The pressure-splitting technique [12] (discussed in section 6.4.1) is applied to solve the discrete linear Stokes system that arises at each stage of the DIRK-CF

Chapter 6. Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

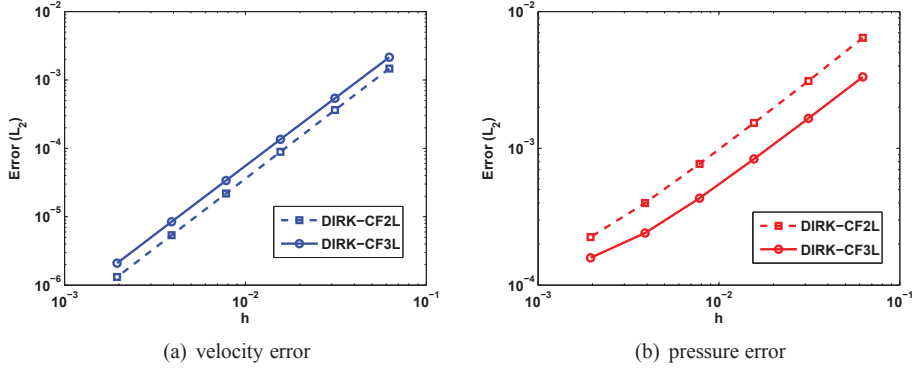


Figure 6.3: Order of convergence of the DIRK-CF2L and DIRK-CF3L. Test problem: Taylor vortex (6.4.3); $Re = 2\pi^2$, $T = 1$, $N = 12$, $N_e = 1$, $\Omega = [-1, 1]^2$, $h = \Delta t = T/2^k$, $k = 4, \dots, 9$. bc: periodic. **(a)** velocity error: DIRK-CF2L (slope = 2.0243), DIRK-CF3L (slope = 2.000); **(b)** pressure error: DIRK-CF2L (slope = 0.9734), DIRK-CF3L (slope = 0.8919).

method. The semi-Lagrangian schemes associated to the DIRK-CF method are achieved by tracking characteristics and interpolating as in [15].

The results reported in both sections 6.4.4 and 6.4.5 indicate that the semi-Lagrangian exponential integrators permit the use of large time step sizes and Courant numbers.

6.4.4 Lid-driven cavity flow in 2D

We consider the 2D lid-driven cavity problem on a domain $(x_1, x_2) \in \Omega := [0, 1]^2$ with initial data $\mathbf{u} = (u, v) = (0, 0)$ and constant Dirichlet boundary conditions

$$u = \begin{cases} 1 & \text{on upper portion of } \partial\Omega \\ 0 & \text{elsewhere on } \partial\Omega \end{cases}, \quad v = 0 \text{ on } \partial\Omega. \quad (6.4.5)$$

In Figure 6.4 (cf. page 127) we demonstrate the performance of the second order DIRK-CF method (SL2L, by the nomenclature of [7]). Spectral element method (see [13]) on a unit square domain $[0, 1]^2$ with $N_e = 10 \times 10$ uniform rectangular elements and polynomial degree $p = 10$ is used. The specified Reynolds numbers considered are $Re = 400, 3200$. A constant time step size, $\Delta t = 0.03$, is used, corresponding to a Courant number of $Cr \approx 9.0911$.

The time integration steps are done until a steady-state solution is attained. A pressure-splitting projection technique [12] is applied to solve each linear Stokes system arising at each stage of the DIRK-CF method. The semi-Lagrangian exponentials in the DIRK-CF

method are solved by tracking characteristics and interpolating as in [15]. In Figure 6.4c-d we plot the streamline contours of the stream functions, choosing contour levels as in [5]. Meanwhile in Figures 6.4a-b plots of the centerline velocities (continuous line, for $Re = 400$, dashed line, for $Re = 3200$) show a good match with those reported in [16] (plotted in red circles). The results in Figure 6.5 show the evolution of the center velocity (at $Re = 400$) up to steady state. It can be observed from this figure that steady state is attained at time $t \approx 40$. At steady state the relative error (L_2 -norm) between the velocity at a given time and the velocity at the preceding time has decreased to $O(10^{-8})$. The results also match with those of [23].

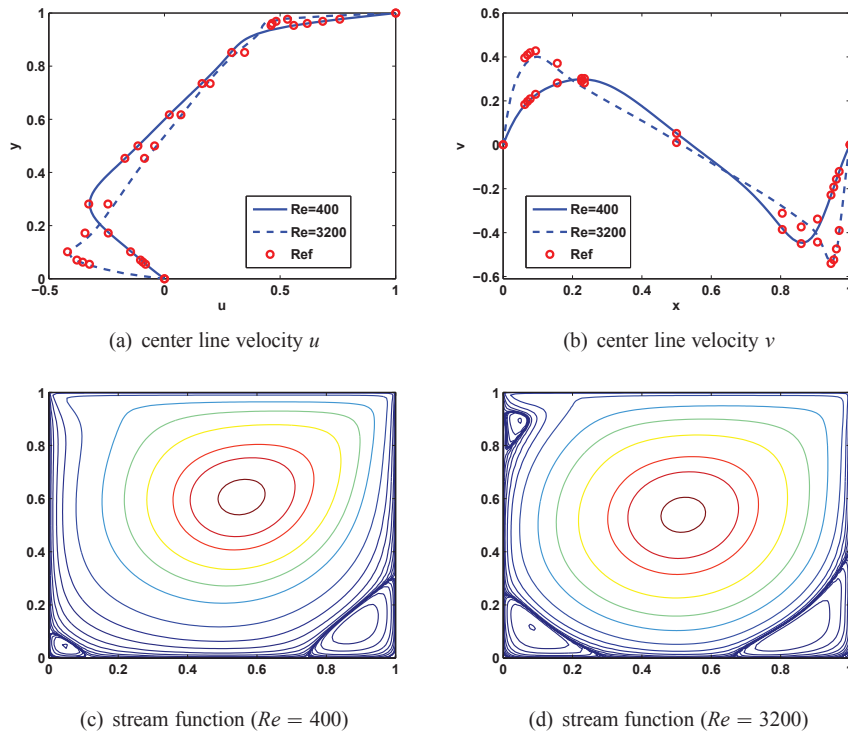


Figure 6.4: Results of a second order DIRK-CF method for the 2D lid-driven cavity problem. We have $(x_1, x_2) \in [0, 1]^2$; $N_e = 10 \times 10$, $N = 10$, $\Delta t = 0.03$, $Cr = 9.0911$. In blue continuous line (our numerical solution); in red circles (\circ , reference solution [16]). **(a)** Horizontal velocity component u along the vertical center line ($x = 0.5$), **(b)** Vertical velocity component v along the horizontal center line ($y = 0.5$), **(c)** Streamline contours of the solution for $Re = 400$, **(d)** Streamline contours of the solution for $Re = 3200$.

Chapter 6. Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

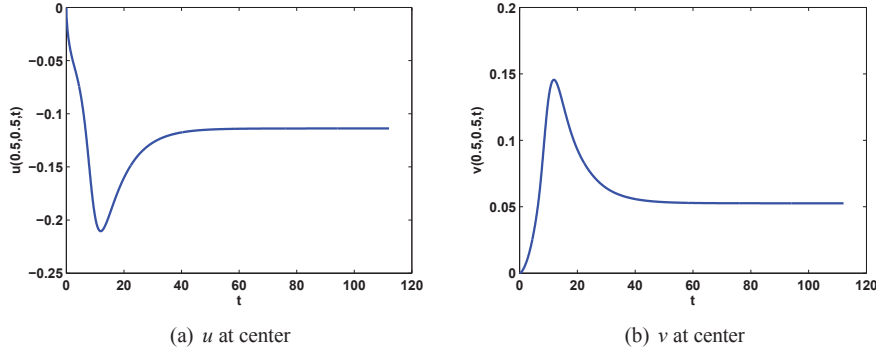


Figure 6.5: Results of a second order DIRK-CF method for the 2D lid-driven cavity problem. We have $(x_1, x_2) \in [0, 1]^2$; $N_e = 10 \times 10$, $p = 10$, $\Delta t = 0.03$, $Cr = 9.0911$, $Re = 400$. **(a)** Evolution of the horizontal velocity component u at the domain center $(x = 0.5, y = 0.5)$: $t \in (0, 112.08)$, **(b)** Evolution of the vertical velocity component v at the domain center $(x = 0.5, y = 0.5)$: $t \in (0, 112.08)$.

6.4.5 Shear layer roll-up problem

We now consider the shear layer problem [4, 11, 13] on a domain $\Omega := [0, 1]^2$ with initial data $\mathbf{u} = (u, v)$ given by

$$u = \begin{cases} \tanh(\rho(x_2 - 0.25)) & \text{for } x_2 \leq 0.5 \\ \tanh(\rho(0.75 - x_2)) & \text{for } x_2 > 0.5 \end{cases}, \quad v = 0.05 \sin(2\pi x_1) \quad (6.4.6)$$

which corresponds to a layer of thickness $\mathcal{O}(1/\rho)$. Doubly-periodic boundary conditions are applied.

In Figure 6.6 we demonstrate the performance of various second order methods including two DIRK-CF methods (SL2 & SL2L, by the nomenclature of [7]) and a second order semi-Lagrangian multistep exponential integrator (named BDF2-CF2, in [8]). The results are obtained at time $t = 1.5$, using a *filter-based* spectral element method (see [13]) with $N_e = 16 \times 16$ elements and polynomial degree $N = 8$. The specified Reynolds number is $Re = 10^5$, while $\rho = 30$ and time step sizes used are $\Delta t = 0.002, 0.005, 0.01$ corresponding to a Courant numbers of $Cr \approx 0.6393, 1.5981, 3.1963$ respectively. The filtering parameter used in each experiment is $\alpha = 0.3$ (see for example [13]). However, the time step size and Courant number are up to about 10 times larger than that report in [13]. The initial values for the BDF2-CF are computed accurately using the second order DIRK-CF (SL2L) with smaller steps. The results are qualitatively comparable with those in [11, 13].

In Figure 6.7 we demonstrate the performance of the second order DIRK-CF method (SL2L). The results are obtained at times $t = 0.8, 1.0, 1.2$ and 1.5 respectively, using

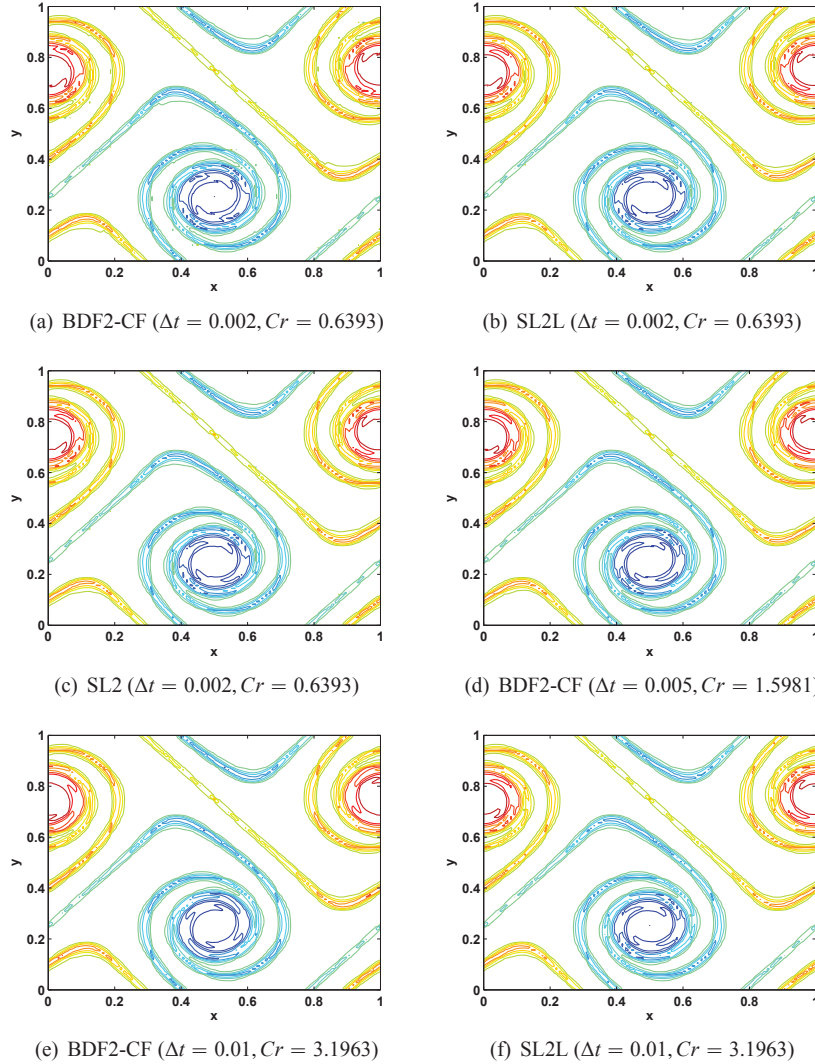


Figure 6.6: Results of second order DIRK-CF methods (SL2 & SL2L) and BDF2-CF method for the shear layer roll-up problem. We have $(x_1, x_2) \in [0, 1]^2$; $N_e = 16 \times 16$, $N = 8$. (filtering, $\alpha = 0.3$), $\rho = 30$, $Re = 10^5$. Vorticity contours (-70 to 70 by 15) of the solution at time $t = 1.5$.

spectral element method (without filtering) with $N_e = 16 \times 16$ elements and polynomial degree $N = 16$. The specified Reynolds number is $Re = 10^5$, while $\rho = 30$. The time step

Chapter 6. Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

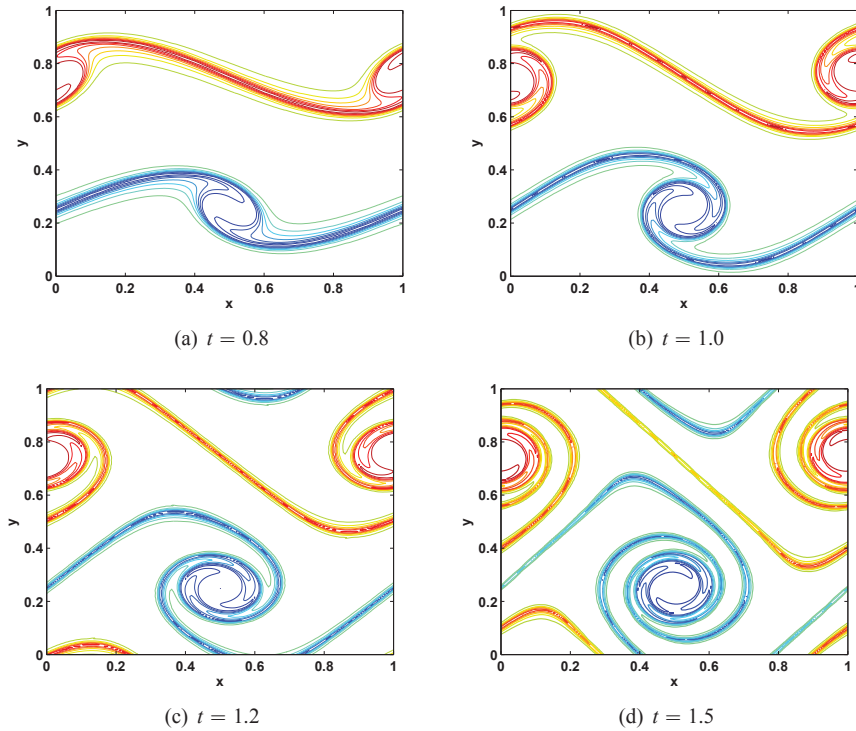


Figure 6.7: Results of second order DIRK-CF method (SL2L) for the shear layer roll-up problem. We have $(x_1, x_2) \in [0, 1]^2$; $N_e = 16 \times 16$, $p = 16$, $\Delta t = 0.01$, $Cr = 11.9250$, $\rho = 30$, $Re = 10^5$. Vorticity contours (-70 to 70 by 15) of the solution at time **(a)** $t = 0.8$, **(b)** $t = 1.0$, **(c)** $t = 1.2$, **(d)** $t = 1.5$.

size used is $\Delta t = 0.01$, corresponding to a Courant number of $Cr \approx 11.9250$. This time step size is 10 times larger than that report in [13]. Again the results are well comparable to those in [11, 13].

Finally in Figure 6.8 we demonstrate the performance of the second order DIRK-CF method (SL2L) for the “thin” shear layer roll-up problem, so defined for $\rho = 100$. The results are obtained at times $t = 0.8, 1.0, 1.2$ and 1.5 respectively, using spectral element method (without filtering) with $N_e = 16 \times 16$ elements and polynomial degree $N = 16$. The specified Reynolds number is $Re = 40,000$. The time step size used is $\Delta t = 0.01$, corresponding to a Courant number of $Cr \approx 11.9250$. The results are well comparable to those in [11, 13], except that we used 10 times the step size in time.

The results reported both in this section and section 6.4.4 indicate that the semi-Lagrangian exponential integrators permit the use of large time step sizes and Courant

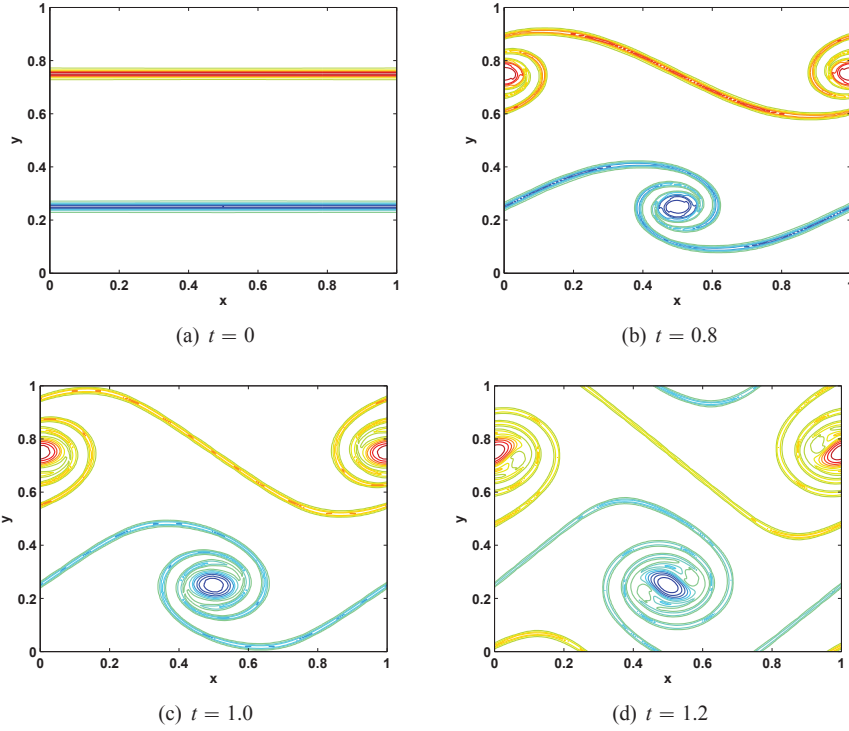


Figure 6.8: Results of second order DIRK-CF method (SL2L) for the “thin” shear layer roll-up problem. We have $(x_1, x_2) \in [0, 1]^2$; $N_e = 16 \times 16$, $p = 16$, $\Delta t = 0.01$, $Cr = 11.9250$, $\rho = 100$, $Re = 40,000$. Vorticity contours (-36 to 36 by 13) of the solution at time (a) $t = 0$, (b) $t = 0.8$, (c) $t = 1.0$, (d) $t = 1.2$.

numbers.

Conclusion

We have derived projection methods based on IMEX Runge-Kutta schemes and semi-Lagrangian exponential integrators (DIRK-CF) for the incompressible Navier-Stokes equations. These methods have been shown to perform well in the case of periodic and no-slip boundary conditions. Using model problems in 2D with high Reynolds number, we have demonstrated the performance of the DIRK-CF methods for convection-dominated problems. The IMEX methods show up to third order of convergence in the velocity. However, the DIRK-CF methods only show up to second order. Proper ways of implementing the projections alongside the boundary conditions for the DIRK-CF

Chapter 6. Semi-Lagrangian exponential integrators for the incompressible Navier-Stokes equations

methods are still to be investigated further. We believe this would help recover the full order of the methods.

References

- [1] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri, *Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations*, Appl. Numer. Math. **25** (1997), 151–167.
- [2] U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton, *Implicit-explicit methods for time-dependent partial differential equations*, SIAM J. Numer. Anal. **32** (1995), no. 3, 797–823.
- [3] D. L. Brown, R. Cortez, and M. L. Minion, *Accurate projection methods for the incompressible Navier-Stokes equations*, J. Comput. Phys. **168** (2001), no. 2, 464–499. MR 1826523 (2002a:76112)
- [4] D. L. Brown and M. L. Minion, *Performance of under-resolved two-dimensional incompressible flow simulations*, J. Comput. Phys. **122** (1995), no. 1, 165–183. MR 1358529 (96g:76038)
- [5] C.-H. Bruneau and M. Saad, *The 2d lid-driven cavity problem revisited*, Computers & Fluids **35** (2006), no. 3, 326–348.
- [6] E. Celledoni, *Eulerian and semi-Lagrangian commutator-free exponential integrators*, CRM Proceedings and Lecture Notes **39** (2005), 77–90.
- [7] E. Celledoni and B. K. Kometa, *Semi-Lagrangian Runge-Kutta exponential integrators for convection dominated problems*, J. Sci. Comput. **41** (2009), no. 1, 139–164.
- [8] ———, *Semi-Lagrangian multistep exponential integrators for index 2 differential-algebraic systems*, J. Comput. Phys. **230** (2011), no. 9, 3413–3429. MR 2780470
- [9] A. J. Chorin, *Numerical solution of the Navier-Stokes equations*, Math. Comp. **22** (1968), 745–762. MR 0242392 (39 #3723)
- [10] Alexandre Joel Chorin, *On the convergence of discrete approximations to the Navier-Stokes equations*, Math. Comp. **23** (1969), 341–353. MR 0242393 (39 #3724)
- [11] P. Fischer and J. Mullen, *Filter-based stabilization of spectral element methods*, C. R. Acad. Sci. Paris Sér. I Math. **332** (2001), no. 3, 265–270. MR 1817374 (2001m:65129)
- [12] P. F. Fischer, *An overlapping Schwarz method for spectral element solution of the incompressible Navier-Stokes equations*, J. Comput. Phys. **133** (1997), no. 1, 84–101. MR 1445173 (97m:76094)

-
- [13] P. F. Fischer, G. W. Kruse, and F. Loth, *Spectral element method for transitional flows in complex geometries*, J. Sc. Comput. **17** (2002), no. 1–4, 81–98.
- [14] C. Foias, O. Manley, R. Rosa, and R. Temam, *Navier-Stokes equations and turbulence*, Encyclopedia of Mathematics and its Applications, vol. 83, Cambridge University Press, Cambridge, 2001. MR 1855030 (2003a:76001)
- [15] P.F. Fischer F.X. Giraldo, J.B. Perot, *A spectral element semi-lagrangian (SESL) method for the spherical shallow water equations*, J. Comput. Phys. **190**.
- [16] U. Ghia, K. N. Ghia, and C. T. Shin, *High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method*, Journal of Computational Physics **48** (1982), no. 3, 387–411.
- [17] J. L. Guermond, P. Mineev, and Jie Shen, *An overview of projection methods for incompressible flows*, Comput. Methods Appl. Mech. Engrg. **195** (2006), no. 44-47, 6011–6045. MR 2250931 (2007g:76157)
- [18] J. L. Guermond and J. Shen, *A new class of truly consistent splitting schemes for incompressible flows*, J. Comput. Phys. **192** (2003), no. 1, 262–276. MR 2045709 (2005k:76076)
- [19] G. E. Karniadakis, M. Israeli, and S. A. Orszag, *High-order splitting methods for the incompressible Navier-Stokes equations*, J. Comput. Phys. **97** (1991), no. 2, 414–443.
- [20] R. Rannacher, *On Chorin’s projection method for the incompressible Navier-Stokes equations*, Lecture Notes in Mathematics, vol. 1530, 1991.
- [21] J. Shen, *On error estimates of projection methods for Navier-Stokes equations: first-order schemes*, SIAM J. Numer. Anal. **29** (1992), no. 1, 57–77. MR 1149084 (92m:35213)
- [22] R. Témam, *Sur l’approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires. II*, Arch. Rational Mech. Anal. **33** (1969), 377–385. MR 0244654 (39 #5968)
- [23] D. Xiu and G. E. Karniadakis, *A semi-Lagrangian high-order method for Navier-Stokes equations*, J. Comput. Phys. **172** (2001), no. 2, 658–684. MR 1857617 (2002g:76077)