

Eulerian and Semi-Lagrangian schemes based on commutator free exponential integrators

Elena Celledoni

ABSTRACT. We consider commutator free exponential integrators combined with splitting and applied to convection diffusion problems semidiscretized in space. Numerical results show improved accuracy of the new approach compared to other methods. Numerical dispersion is however experienced due to the classical Eulerian formulation of the problem. A semi-Lagrangian approach is then proposed to take care of this drawback.

1. Introduction

Commutator-free exponential integrators were proposed in [CMO]. These are a new class of Lie group integrators which do not require the use of commutators for achieving numerical approximations of order higher than 2. In [CMO] the good performance of these methods in dealing with stiff problems has been substantiated by numerical tests. Krogstad in [K2] shows the relationship between the commutator free exponential integrators and the methods developed by Cox and Matthews in [CMt], and by Kassam and Trefethen in [KT], and the good properties of the commutator free methods in dealing with stiffness is further investigated and explained. Another class of exponential integrators for large stiff systems of differential equations was introduced by Hochbruck and Lubich [HL], and Hochbruck Lubich and Selhofer [HLS]. These methods are also 'commutator-free', but their relationship with the classes of methods mentioned above remains to be clarified.

In this paper we focus on the use of commutator-free exponential integrators on convection diffusion problems of the type

$$(1.1) \quad \frac{\partial}{\partial t} u(\mathbf{x}, t) + \mathbf{V} \cdot \nabla u(\mathbf{x}, t) = \Gamma \nabla^2 u + f(\mathbf{x})$$

with $\mathbf{x} \in \Omega \subset \mathbb{R}^d$ and $\mathbf{V} : \mathbb{R}^d \times [0, T] \rightarrow \mathbb{R}$ is a given vector field, $u : \mathbb{R}^d \times [0, T] \rightarrow \mathbb{R}^d$, and $u(\mathbf{x}, 0) = u_0(\mathbf{x})$. We can distinguish between two important special cases of the above equation. The first case arises when u is a conserved passive scalar, $f(x) = 0$, u can represent temperature, or salt concentration in the water, \mathbf{V} is known a priori. The second case is given by the Navier-Stokes equations where u is a vector field, $\mathbf{V} = u$, and $f(x) = -\nabla p$, with the constraint $\nabla \cdot u = 0$.

1991 *Mathematics Subject Classification*. Primary 54C40, 14E20; Secondary 46E25, 20C20.

Key words and phrases. Lie group methods, convection-diffusion equations, semi-Lagrangian methods.

The first author was supported in part by NSF Grant #000000.

The source of stiffness in these problems is due to the discretization of the diffusion. If we semidiscretize (1.1) in space with a finite element or finite difference method we obtain a system of ordinary differential equations of the type

$$By_t - Cy = Ay, \quad y(0) = y_0,$$

here C is the discretized convection operator which in some cases is a skew-symmetric matrix, A corresponds to the linear diffusion term often symmetric and negative definite, and B a mass matrix which we assume symmetric and positive definite and we get

$$(1.2) \quad y_t - B^{-1}Cy = B^{-1}Ay, \quad y(0) = y_0.$$

Note that if C is skew symmetric then $B^{-1}C$ is skew-symmetric with respect to the inner product $\langle \cdot, B \cdot \rangle$ where $\langle \cdot, \cdot \rangle$ is the Euclidean inner product. In general the discretized convection operator can be nonlinear, $C = C(y)$. A typical approach for solving numerically the semidiscretized equations is to treat convection and diffusion separately, the diffusion with an implicit approach and the convection with an explicit integrator possibly of higher order, see for example [ARW], [CHQZ]. We here will follow the approach proposed in [MPR] by Maday, Patera and Rønquist. In this paper the authors propose a method called Operator Integrating Factor (OIF), based on a splitting technique, this method coincides with other more classical Integrating Factor approaches [T] in the linear case. Let us consider the following change of variables

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

with

$$(1.3) \quad W_t = B^{-1}CW, \quad W(0) = I,$$

after differentiation we obtain the following equation for z

$$(1.4) \quad z_t = W^{-1}B^{-1}AWz, \quad z(0) = y_0,$$

note that if $B^{-1}C$ is skew-symmetric with respect to some suitable inner product, then W has columns mutually orthonormal with respect to the same inner product.

The numerical solution of the transformed equation, (1.4), is then carried out and the numerical approximation is transformed back by applying $W(t)$. The matrix solution W of (1.3) is never computed explicitly, but rather one computes directly the result of applying $W(t)$ to some vector g , approximating then a semidiscretized pure convection problem of the type

$$(1.5) \quad w' = B^{-1}Cw, \quad w(0) = g.$$

In the case C is not depending on y and \mathbf{V} is known and constant in time then $W = \exp(tB^{-1}C)$, even if \mathbf{V} is time dependent the solution of (1.3) is decoupled from the solution of (1.4). In the nonlinear case the two equations must instead be solved coupled.

One advantage of using such a change of variables can be seen in the following example, suppose we apply a discretization method, e.g. the implicit Euler scheme directly to (1.2), where we assume for simplicity that both diffusion and convection are linear, then we will need to solve at each time step a non-symmetric linear system of equations of the type

$$(I - hB^{-1}(A + C))y_1^{IE} = y_0$$

while the same method applied to the reformulated equation (1.4) gives the symmetric and positive definite system

$$(I - he^{-hB^{-1}C} B^{-1} A e^{hB^{-1}C}) z_1^{IE} = z_0.$$

It is well known that there is a computational advantage in solving symmetric positive definite systems compared to general ones in terms of iterative techniques available.

In this paper we will restrict our attention to the linear case although generalizations to the case of nonlinear convection are possible.

We will here combine the approach of [MPR] with the use of commutator free exponential integrators for PDEs as proposed in [CMO]. In [MPR] the equation (1.4) for z is solved implicitly, while we propose to treat this equation explicitly with the commutator free methods. Our approach is competitive with the methods described in [MPR] and based on a *backward difference formula* for the implicit discretization of (1.4), or their variant obtained by using implicit Runge-Kutta integration of (1.4) using a Radau I A method. The classical implicit methods require typically the solution of symmetric and positive definite algebraic systems of equations involving the matrix $(I - hB^{-1}A)$ while the commutator-free methods require the computation of exponentials of the type $\exp(hB^{-1}A)v$, where h is the time step used for the integration. It is well known that using Krylov subspace methods for solving each of the two linear algebra problems results in a faster (superlinear) convergence in the case of the exponential. This at least in the case the linear system is not successfully preconditioned. We refer to [HL], [CM] and references therein for precise convergence estimates for Krylov subspace methods in the two cases.

Finally we propose a semi-Lagrangian version of the same method which shows higher performance in dealing with numerical dispersion. These methods are obtained by substituting the operator $W(t)$ in (1.4), with its approximation $\tilde{W} = W + E$ obtained by computing the numerical solution of the undiscretized pure convection problem corresponding to (1.5). This is done directly by computing the characteristics exiting from the grid points of the discretization.

2. Commutator-free methods

Consider the following differential equation

$$y' = F(y), \quad y_0 \in \mathbb{R}^n,$$

F is a vector field on \mathbb{R}^n .

Assume there exist a set of frame vector fields,

$$E_1, E_2, \dots, E_d,$$

such that $\forall y \in \mathbb{R}^n$

$$F(y) = \sum_{i=1}^d f_i(y) E_i(y)$$

and $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$. In the commutator free exponential integrators we consider linear combinations of vector fields of the type

$$F_p(y) = \sum_{i=1}^d f_i(p) E_i(y),$$

and $F_p \in \text{span}\{E_1, \dots, E_d\}$, and we say that F_p is the vector field F *frozen* at the point p .

For a given vector field F we denote with $\exp(hF)p$ the solution of the differential equation

$$y' = F(y), \quad y(0) = p,$$

on the interval $[0, h]$.

A commutator-free exponential integrator for the considered equation has then the following format:

Commutator-free method

for $r = 1 : s$ do

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

$$F_r = hF_{Y_r} = h \sum_i f_i(Y_r) E_i$$

end

$$y_1 = \exp(\sum_k \beta_J^k F_k) \cdots \exp(\sum_k \beta_1^k F_k)p$$

The integrator has s stages and parameters α_{rJ}^k, β_J^k . Each new stage value is obtained as a composition of J exponentials 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.

$$\begin{array}{c|cc} 0 & & \\ \frac{1}{3} & \frac{1}{3} & \\ \frac{2}{3} & 0 & \frac{2}{3} \\ \hline & \frac{1}{3} & 0 & 0 \\ & -\frac{1}{12} & 0 & \frac{3}{4} \end{array} \quad \begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ \frac{1}{2} & 0 & \frac{1}{2} & \\ 1 & \frac{1}{2} & 0 & 0 \\ & -\frac{1}{2} & 0 & 1 \\ \hline & \frac{1}{4} & \frac{1}{6} & \frac{1}{6} & -\frac{1}{12} \\ & -\frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{4} \end{array}$$

We want to apply these methods to (1.4), and then obtain numerical approximations for (1.2) by using the transformation $y_n = W(t_n)z_n$, assuming linear convection constant in time this means $W(t) = \exp(tB^{-1}C)z_n$ and $W(t)^{-1} = \exp(-tB^{-1}C)z_n$. The considered vector field is then

$$F(z) = W(t)^{-1}B^{-1}AW(t)z,$$

which we rewrite in a shorter format as follows

$$F(z) = W(t)^{-1}\tilde{A}W(t)z,$$

where $\tilde{A} = B^{-1}A$.

Instead of applying the methods in a straightforward manner we here consider a new *ad hoc* and computationally efficient format for the methods. This format

exploits the linearity of the right hand side of equation (1.4), and reduces the number of times one has to apply $W(t)$.

The vector fields frozen at the stage values are

$$F_{Z_r}(t, z) = W(t_0 + c_r h)^{-1} \tilde{A} W(t_0 + c_r h).$$

The exponential of any such vector field is the composition of three operators and it is of the type,

$$\exp(\alpha h F_{Z_r}) = W(t_0 + c_r h)^{-1} \exp(\alpha h \tilde{A}) W(t_0 + c_r h).$$

The stage values are not directly involved in the computation of the frozen vector fields and for the method of order three for example we have

$$\begin{aligned} F_1 &= hW(t_0)^{-1} \tilde{A} W(t_0) \\ F_2 &= hW((t_0 + h/3))^{-1} \tilde{A} W(t_0 + h/3) \\ F_3 &= hW(t_0 + 2h/3)^{-1} \tilde{A} W(t_0 + 2h/3). \end{aligned}$$

The advancement of the solution is obtained by the composition of two exponentials, the first exponential corresponds to the first of the two rows of the β values and applied to the initial value gives the second stage Z_2

$$Z_2 = W(t_0)^{-1} \exp(h/3 \tilde{A}) W(t_0) \cdot p,$$

and the second exponential, corresponding to the second row of β values, arises from the linear combination of two frozen vector fields,

$$\begin{aligned} -h/12F_1 + 3/4hF_3 &= \\ &= -h/12W(t_0)^{-1} \tilde{A} W(t_0) + \\ &= 3/4hW(t_0 + 2/3h)^{-1} \tilde{A} W(t_0 + 2/3h) \\ &= W(t_0)^{-1} (-h/12\tilde{A} + 3/4hW(2/3h)^{-1} \tilde{A} W(2/3h)) W(t_0). \end{aligned}$$

The numerical solution at $t_1 = t_0 + h$ is then

$$z_1 = \exp(-h/12F_1 + 3/4hF_3) \cdot Z_2$$

and the corresponding numerical solution y_1 for the equation (1.1) is $y_1 = W(t_1)z_1$, and then

$$y_1 = W(h) \exp(-h/12\tilde{A} + 3/4hW(2/3h)^{-1} \tilde{A} W(2/3h)) \cdot Y_2$$

where $Y_2 = \exp(h/3\tilde{A})W(t_0)y_0$.

In summary we get to the following format for the implementation of the commutator free third order method for the numerical solution of (1.2),

Order 3 CF

```

p = y0
for n = 0 : N - 1 do
    Y = exp(h/3\tilde{A})p
    w_{n+1} = exp(-h/12\tilde{A} + 3/4hW(2/3h)^{-1}\tilde{A}W(2/3h)) \cdot Y
    y_{n+1} = W(h)w_{n+1}
    p = y_{n+1}
end
    
```

By analogous considerations we arrive to a similar expression for the implementation of the fourth order method.

Order 4 CF

```

p = y0
for n = 0 : N - 1 do
  Z = W(h)p
  Y = exp(h/4W(h)ÃW(h)-1 + h/3W(h/2)ÃW(h/2)-1 - h/12Ã) · Z
  yn+1 = exp(-h/12W(h)ÃW(h)-1 + h/3W(h/2)ÃW(h/2)-1 + h/4Ã) · Y
  p = yn+1
end

```

The main cost of these methods is due to the computation of the exponentials giving w_{n+1} in the order 3 method and computing Y and y_{n+1} in the order 4 method. We assume these exponentials are computed via Krylov subspace techniques which require only the repeated application of the operator to be exponentiated to a suitable vector.

As comparison methods we consider two *backward difference formulas* (BDF) of order 3 and 4 respectively. These methods are implicit as opposed to the methods described above, but still particularly for our linear problem they require the inversion of just one linear system of equations per step. The BDF methods are multistep methods and as such they require the preliminary computation of the starting values $y_0 \approx y(0), \dots, y_k \approx y(kh)$, with $k-1$ the order of the BDF method, this is usually done by using a one step method. In our case we used the values of the exact solution as starting values for the BDF methods.

The BDF3 as applied to (1.4) gives rise to the following difference equation

$$W(t_n)^{-1}(I - 6/11\tilde{A})W(t_n)z_n = 18/11z_{n-1} - 9/11z_{n-2} + 2/11z_{n-3}$$

and then transforming the equation in the y variables we get

$$(I - 6/11\tilde{A})y_n = 18/11W(h)y_{n-1} - 9/11W(2h)y_{n-2} + 2/11W(3h)y_{n-3}.$$

The BDF4 has instead the following format

$$(I - 12/25h\tilde{A})y_n = b$$

$$b = 48/25W(h)y_{n-1} - 36/25W(2h)y_{n-2} + 16/25W(3h)y_{n-3} - 3/25W(4h)y_{n-4}.$$

Since the CF methods are one step methods we consider for comparison also a Runge-Kutta method of order 3, the Radau I A method which is A-stable and implicit, ([HW], p.73). By exploiting the linearity of the vector field this method is implemented as follows.

Radau I A, order 3

```

p = y0
for n = 0 : N - 1 do
  An = W(tn)-1ÃW(tn)
  A2/3 = W(2/3h)-1ÃW(2/3)
  (I - h(5/12I - h/16An(I - h/4An)-1)A2/3)Z = (I - h/4An)-1p
  R = (I - h/4An)-1(p - h/4A2/3Z)
end

```

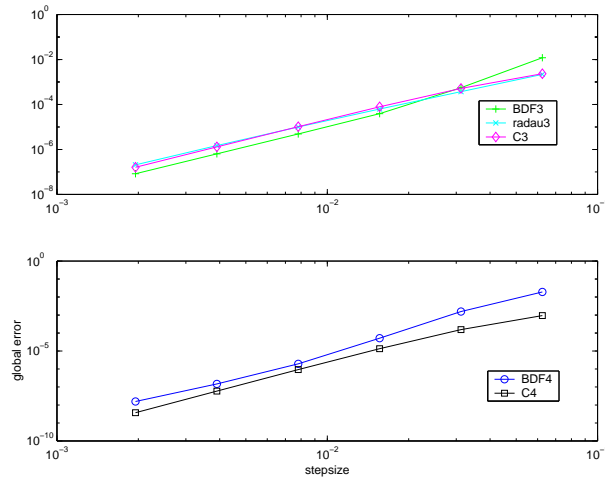


FIGURE 1. Global error versus stepsize for different methods applied to the semidiscretized convection diffusion linear equation: BDF, Commutator-free, Radau I A, $\Gamma = 0.1$

$$z_{n+1} = p + h(1/4A_n R + 3/4A_{2/3} Z)$$

$$p = z_{n+1}$$

$$y_{n+1} = W(t_n)z_{n+1}$$

end

The operators A_n and $A_{2/3}$ are never computed explicitly, but just applied to suitable vectors while solving the linear systems of algebraic equations in the algorithm via iterative methods. These operators depend on A and $W(t)$ like in the CF case.

The results of the numerical experiments reported in Figures 1, 2, 3 show the norm of the global error in the semidiscretized problem (1.2), for the methods, plotted against the stepsize in a logarithmic scale. The experiments have been repeated with different magnitudes of the viscosity $\Gamma = 0.1, 0.01, 0.001$ in (1.1), allowing for different regimes of stiffness in the problem.

We have considered the equation (1.1) in one space dimension, $x \in [-1, 1]$, $u(x, 0) = \text{sech}(20(x - x_0)\pi)$, ($x_0 = 0.5$), $\mathbf{V} = 1$. The discretization is carried out with a spectral element method with 10 elements and polynomials of degree 4, with interpolation on the Gauss Lobatto Legendre nodes. The operator $W(t) \cdot g = \exp(-tB^{-1}C) \cdot g$ and it is computed exactly with the built in function of Matlab, *expm*. The exponentials involving the matrix \tilde{A} in the commutator free methods are computed iteratively using Krylov subspace methods. Also the linear systems of equations for the BDF methods and the Radau method are solved numerically via iterative Krylov subspace techniques.

The interval of integration is $[0, 1]$. The error is the Euclidean norm of the difference of the approximation and the exact solution of the semi-discretized problem.

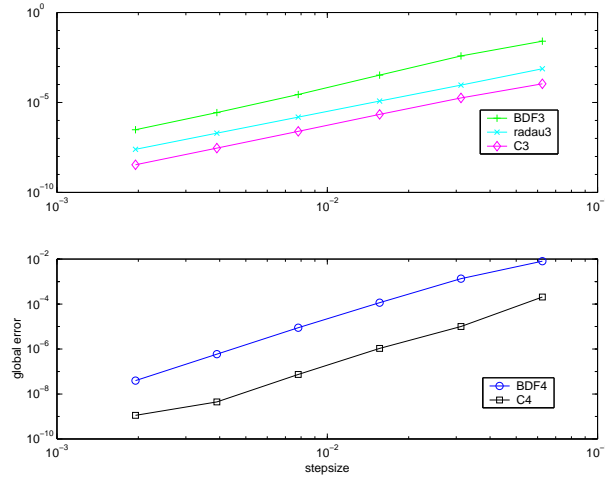


FIGURE 2. Global error versus stepsize for different methods applied to the semidiscretized convection diffusion linear equation: BDF, Commutator-free, Radau I A, $\Gamma = 0.01$

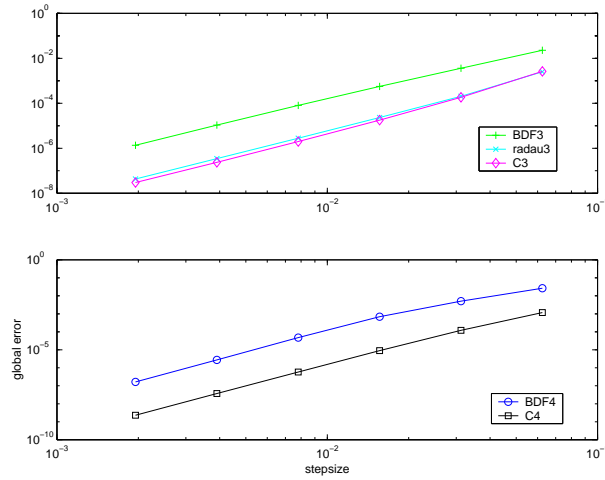


FIGURE 3. Global error versus stepsize for different methods applied to the semidiscretized convection diffusion linear equation: BDF, Commutator-free, Radau I A, $\Gamma = 0.001$

When $\Gamma = 0.01$, the CF methods achieve lower global error for the same size of the time step compared to BDF and Radau. Both Radau and CF, which are based on Runge-Kutta formulae, do better than the BDF methods for $\Gamma = 0.01$ and $\Gamma = 0.001$. As Γ decreased, the performance of the Radau method approaches that of the CF method.

In Figure 4 we plot the values of the global error against the number of flops required for the different methods in our implementation. The commutator-free methods perform competitively with respect to the BDF methods, but we do not

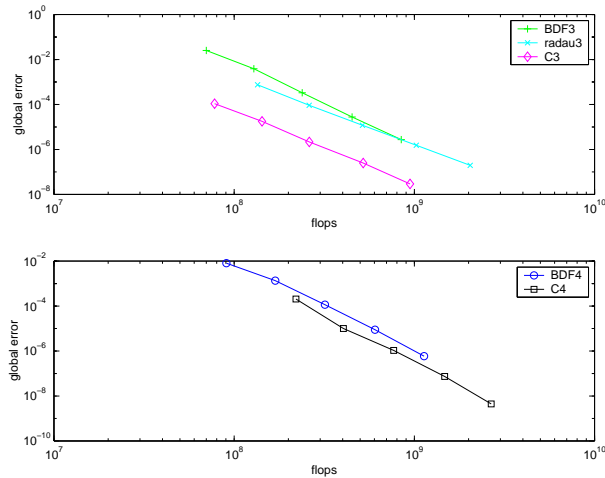


FIGURE 4. Global error versus number of flops for different methods applied to the semidiscretized convection diffusion linear equation: BDF, Commutator-free, Radau I A, $\Gamma = 0.01$

exclude improvements in the performance of the BDF methods if a good and cheap preconditioner is available for the solution of the linear systems.

3. Semi-Lagrangian version of the algorithm

Especially for small values of Γ the quality of the solution of (1.1) is polluted by the presence of oscillations, at least when the spatial discretization is not fine enough. The exact solution of the semidiscretized equation (1.2) does not propagate all the frequencies of the solution of (1.1) at the right speed. We can illustrate the phenomenon with comparing the exact steady state solution of the following convection diffusion equation in one space dimension

$$u_t + u_x = \nu u_{xx} + f, \quad u(x, 0) = \cos(\pi/2x), \quad f = 1,$$

$x \in [-1, 1]$, $u(t, 1) = u(t, -1) = 0$, with the exact solution of its discretized version

$$U_t = B^{-1}(A + C)U + f, \quad U_i(0) = u(x_i, 0)$$

where C is the discrete convection operator and A is the discrete diffusion, obtained by considering a discretization in space with spectral methods on the Gauss Lobatto Legendre nodes, and x_i is a node in the resulting grid, $i = 0, \dots, p + 1$. In the numerical experiments we considered $p = 16$ and $p = 32$.

In this case the forcing term f gives rise to a modification of the format of equation (1.4), which is now

$$z_t = W^{-1}(t)\tilde{A}W(t) + W^{-1}(t)f, \quad z(0) = y_0,$$

the corresponding frozen vector fields are then identified by the couples

$$F_{Z_r}(t, z) = (W(t_0 + c_r h)^{-1}\tilde{A}W(t_0 + c_r h), W(t_0 + c_r h)^{-1}f),$$

and their exponentials are the solutions of the differential equations

$$v_t = W(t_0 + c_r h)^{-1}\tilde{A}W(t_0 + c_r h) + W(t_0 + c_r h)^{-1}f,$$

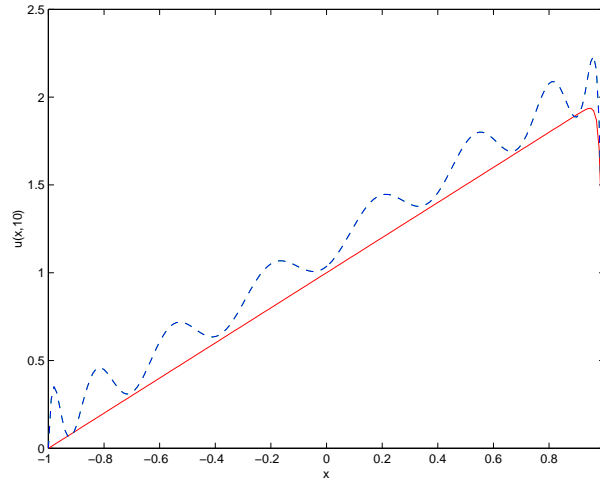


FIGURE 5. Exact solution of the semi-discretized problem (dashed line) and exact solution (solid line) at $t = 10$, $\nu = 0.01$, $p = 16$.

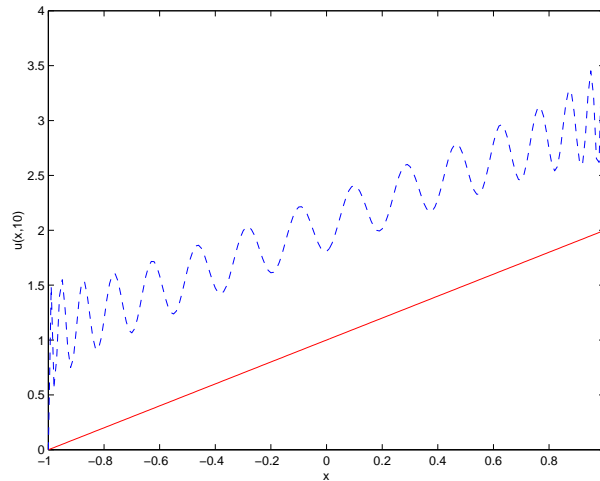


FIGURE 6. Exact solution of the semi-discretized problem (dashed line) and exact solution (solid line) at $t = 10$, $\nu = 0.001$, $p = 32$.

with suitable initial values.

As we can note in Figure 5 and Figure 6 the oscillating exact solution of the discretized problem (dashed line) do not resemble qualitatively the features of the exact solution of the PDE especially when ν is small.

We here consider semi-Lagrangian schemes based on the direct approximation of the pure convection problems (1.5) previously approximated by the exponentials $W(t)g = \exp(-tB^{-1}C)g$. We substitute the operator W with a perturbed operator $\tilde{W} = W + E$ whose construction is discussed here below. This introduces an error in the solution of the discretized equation (1.2), but gives improved performance in the approximation of the PDE problem.

We consider functions $w(x, t)$ such that $w_t + w_x = 0$, $w(x, 0) = w_0(x)$ and we assume $w_0(x_i) = g_i$, the boundary conditions are taken according to the boundary conditions of the original problem, we here consider $w(-1, t) = w(1, t) = 0$. We then have that $w(x, t) = w(x - t, 0)$ and in particular on the nodes of the discretization $w(x_i, t) = w(x_i - t, 0)$. We assume $w \approx w_p(x, t) = \sum_{j=0}^p w_{p,j}(t) l_j(x)$, where $l_j(x)$ are the Lagrange basis functions defined on the nodes of the discretization, and $w_{p,j}(t)$ are the nodal values of the approximation at time t , i.e. $w_{p,j}(t) = w_p(x_j, t)$. Since $w(x, t) = w(x - t, 0)$ we compute

$$w_{p,i}(t) = w_p(x_i - t, 0) = \sum_{j=0}^p w_{p,j}(0) l_j(x_i - t),$$

in other words we obtain the values of the approximation w_p at time t on the nodes of the grid, by interpolating the known values of the initial function on the nodes. If $\mathbf{g}(t)$ is a vector whose $p + 1$ components are the values of $w_p(x, t)$ on the nodes then

$$\mathbf{g}(t) = \tilde{W}(t) \mathbf{g}(0), \quad \tilde{W}_{i,j}(t) = l_j(x_i - t),$$

with $\tilde{W}_{i,j}(t)$ the (i, j) entry of the operator \tilde{W} written in a matrix form. The cost of evaluating each entry of the operator \tilde{W} is $\mathcal{O}(p)$ giving an overall cost of $\mathcal{O}(p^3)$. This cost should be compared with the cost of computing $\exp(tB^{-1}C)\mathbf{g}(0)$ which is $\mathcal{O}(p^2)$ if we use iterative methods for the approximation of the matrix exponential and we have fast convergence. The qualitative performance of the semi-Lagrangian methods is however clearly superior as can be seen in Figures 7 and 8.

For more general convection diffusion problems in several space dimensions and with time dependent convecting vector fields \mathbf{V} , the considered pure convection problems are of the type $u_t + \mathbf{V} \cdot \nabla u = 0$ and we have that

$$\tilde{W}_{i,j}(t) = l_j(X_i(t)),$$

where $X_i(t)$ is the characteristic exiting from the grid point x_i and satisfies the ordinary differential equation

$$\dot{X}_i = V(X_i(t)), \quad X_i(0) = x_i.$$

We report in Figures 7 and 8 the results of applying the semi-Lagrangian version of the algorithm CF3 to the convection diffusion equation given at the beginning of this section. We computed the error as the max norm of the difference between numerical approximation and the exact solution of the original undiscretized problem on the nodes of the discretization. This error is big and dominated by the error of the spatial discretization for the Eulerian version of the methods, while it is much smaller in the semi-Lagrangian case.

In this paper we have considered Commutator-free exponential integrators for convection diffusion problems. We have applied the methods to a reformulated semidiscretized system of ordinary differential equations defined by a symmetric negative semi-definite operator. This problem is obtained after a change of variables which separates in a splitting fashion the semidiscrete convection operator from the semidiscrete diffusion operator. Exponential integrators are well suited for this kind of problems, because the computation of the matrix exponential using Krylov subspace methods is working well in the case of symmetric negative semi-definite problems. The comparison of the CF methods with BDF of order 3 and 4 and Radau I A of order 3, shows the competitiveness of the proposed approach.

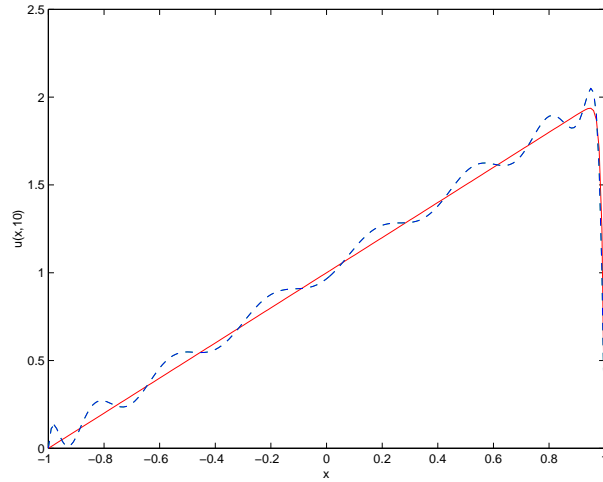


FIGURE 7. Numerical (dashed line) and exact (solid line) solution at $t = 10$, $\nu = 0.01$, $p = 16$.

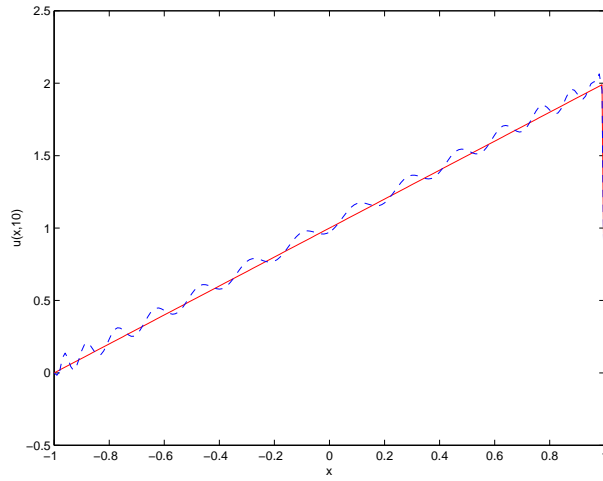


FIGURE 8. Numerical (dashed line) and exact (solid line) solution at $t = 10$, $\nu = 0.001$, $p = 32$.

Improved performance has been obtained by considering a semi-Lagrangian version of the methods arising naturally from the type of splitting considered. The numerical experiments are limited to some simple test problems in one space dimension and the methods will be further tested in more challenging applications.

Acknowledgements The author is grateful to Einar Rønquist, Robert McLachlan and Brynjulf Owren for useful discussions.

References

- [ARW] U. M. Ascher, S. J. Ruuth and B. T. R. Wetton, *Implicit-explicit methods for time-dependent partial differential equations*, SIAM J. Num. Anal., **32** (1995), 797–823.

- [CHQZ] C. Canuto, M. Y. Hussaini, A. Quarteroni and T. A. Zang, *Spectral methods in Fluid Dynamics*, Springer-Verlag, Berlin, (1988).
- [CMO] E. Celledoni and A. Marthinsen and B. Owren, *Commutator-free Lie group methods*, FCGS, **19** (2003), 341–352.
- [CM] E. Celledoni and I. Moret *A Krylov projection method for systems of ODEs*, Appl. Num. Math., **24** (1997), 365–378.
- [CMt] S.M. Cox and P.C. Matthews *Exponential time differencing for stiff systems*, J- of Comp. Phys. , **176** (2002), 430–455.
- [KT] A. K. Kassam and L. N. Trefethen, *Fourth-order time stepping for stiff PDEs*, University of Oxford, (2002), submitted to SIAM J. Sci. Comput.
- [HW] E. Hairer and G. Wanner *Solving Ordinary Differential Equations II*, Springer series in Computational Mathematics, Springer, (1996), second edition.
- [HL] M. Hochbruck and C. Lubich *On Krylov subspace approximations of the matrix exponential operator*, SIAM J. Num. Anal., **34** (1997), 1911–1925.
- [HLS] M. Hochbruck, C. Lubich and H. Selhofer *Exponential integrators for large systems of differential equations*, SIAM J. Sci. Comput., **19** (1998), 1552–1574.
- [K2] S. Krogstad, *Generalized Integrating Factor methods for stiff PDEs* , University of Bergen, (2003), submitted to J. Comp. Phys
- [K1] S. Krogstad, *RKMK-related methods for stiff nonlinear PDEs*, submitted to Appl. Numer. Math., Innovative Time Integrators for PDEs proceedings, (2002).
- [MPR] Y. Maday and A. T. Patera and E. M. Rønquist, *An operator integration factor splitting method for time dependent problems: Application to incompressible fluid flows*, J. of Sci. Comp., **5** (1990), 263–292.
- [T] L. N. Trefethen, *Spectral Methods in Matlab*, SIAM, (2000).

CENTRE FOR ADVANCED STUDY, DRAMMENSVEIEN 78, OSLO, NORWAY

Current address: SINTEF Applied Mathematics, Sem Sælandsvei 5, 7465 Trondheim, Nor-

way

E-mail address: elenac@math.ntnu.no