A second-order diagonally-implicit-explicit multi-stage integration method

Hong Zhang\textsuperscript{a}, Adrian Sandu\textsuperscript{b,}\textsuperscript{*}

\textsuperscript{a}Computational Science Laboratory, Department of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 (zhang@vt.edu)

\textsuperscript{b}Computational Science Laboratory, Department of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 (sandu@cs.vt.edu)

Abstract

Implicit-explicit time stepping methods have proved useful for efficiently solving problems with both stiff and nonstiff components. IMEX Runge-Kutta methods and IMEX multistep methods have been studied in the literature. But there are no available IMEX general linear methods (IMEX-GLMs). In this paper, we construct IMEX schemes based on diagonally-implicit multi-stage integration methods. The new algorithms have great potential for practical use. Numerical results indicate the observed order of accuracy matches the theoretical order.

Keywords: implicit-explicit, general linear methods, diagonally-implicit multi-stage integration methods

1. Introduction

The implicit-explicit (IMEX) time integration schemes have been increasingly used for solving problems with both stiff and nonstiff components, which may arise from a wide range of areas such as mechanical and chemical engineering, astrophysics, meteorology and oceanography, and environmental science. Well-known examples include advection-diffusion-reaction problems, systems containing fluid-membrane interactions, and Navier-Stokes equations. Such problems can all be expressed in a simplified model as

\[ \dot{u} = f(u) + g(u), \]

where \( f \) corresponds to the nonstiff term, and \( g \) corresponds to the stiff term. An IMEX scheme treats the nonstiff term explicitly while handles the stiff term implicitly, thereby gaining the benefits from the low cost of explicit methods and favorable stability properties of implicit methods.

IMEX linear multistep methods have been developed in [1, 2, 3], and IMEX Runge-Kutta methods have been built in [4, 5, 6, 7]. These two families of IMEX methods have been accommodated into a generic framework based on the unifying concept of general linear methods (GLMs), which was proposed by J.C. Butcher [8]. This framework allows easy switch between various explicit and implicit time-stepping methods and finally facilitate to find more efficient methods and time-discretization verification studies.
However, Runge-Kutta methods and linear multistep methods are only special cases of GLMs. There are a large number of other possibilities for acceptable methods, among which diagonally implicit multistage integration methods (DIMSIM) [9] are identified to be efficient and accurate and to have great potentials for practical use. The added complexity of GLMs such as DIMSIM improves the flexibility to develop methods with better stability and accuracy properties. To the best of our knowledge, there is no available IMEX schemes based on GLMs.

In this study, we propose a new family of implicit-explicit methods based on pairs of DIMSIMs, as an effort toward ambitious accommodation of all existing general linear methods into one unified framework.

2. General Linear Methods

2.1. Representation of General Linear Methods

Consider the initial value problem for an autonomous system of differential equations in the form

\[ y'(t) = f(y(t)), \quad t \in [t_0, t_F], \]
\[ y(t_0) = y_0, \]  

(2)

\( f : \mathbb{R}^m \to \mathbb{R}^m, y_0 \in \mathbb{R}^m \). GLMs [10] for (2) can be represented by the abscissa vector \( c = [c_1, c_2, \ldots, c_s]^T \), and a table of four coefficient matrices

\[
\begin{array}{c|c}
A & U \\
B & V
\end{array}
\]

\( A \in \mathbb{R}^{s \times s}, U \in \mathbb{R}^{s \times r}, B \in \mathbb{R}^{r \times s} \) and \( V \in \mathbb{R}^{r \times r} \). On the uniform grid \( t_n = t_0 + nh, n = 0, 1, \ldots, N, Nh = T - t_0 \), the fixed step size version of these methods takes the form

\[
Y_i = h \sum_{j=1}^{s} a_{ij} f(Y_j) + \sum_{j=1}^{r} u_{ij} y_{j}^{[n-1]}, \quad i = 1, 2, \ldots, s, \]  

(3a)

\[
y_i^{[n]} = h \sum_{j=1}^{s} b_{ij} f(Y_j) + \sum_{j=1}^{r} v_{ij} y_{j}^{[n-1]}, \quad i = 1, 2, \ldots, r, \]  

(3b)

where \( s \) is the number of internal stages and \( r \) is the number of external stages. Here, \( h \) is the step size, \( Y_i \) is an approximation to \( y(t_n-1 + c_i h) \) and \( y_i^{[n]} \) is an approximation to the linear combination of the derivatives of \( y \) at the point \( t_n \). The corresponding vector form is

\[
Y = h(A \otimes I)F(Y) + (U \otimes I)y_{0}^{[n-1]}, \]  

(4a)

\[
y^{[n]} = h(B \otimes I)F(Y) + (V \otimes I)y_{0}^{[n-1]}, \]  

(4b)

where \( I \) is an identity matrix of the same dimension of the ODE system (m).

The stability function \( p(w, z) \) is defined by the formula

\[
p(w, z) = det(wI - M(z)), \]  

(5)

\( w \in \mathbb{C} \) and \( M \) is the stability matrix defined by

\[
M(z) = V + zB(I - zA)^{-1}U. \]  

(6)

2.2. Diagonally implicit multistage integration methods

Diagonally implicit multistage integration methods (DIMSIMs), as a subclass of GLMs, was proposed by Butcher [8] to overcome the limitations of linear multistep methods and Runge-Kutta methods (lack of A-stability of high order linear multistep methods and low stage order for implicit Runge-Kutta methods). This class of methods is characterized by the following properties:

1. \( A \) is lower triangular with the same element \( \lambda \) on the diagonal;
2. \( V \) is a rank 1 matrix with nonzero eigenvalue equal to 1 to guarantee preconsistency;
3. Order \( p \), stage order \( q \), number of external stages \( r \), number of internal stages \( s \) are related by \( p = q \) or \( p = q + 1 \) and \( r = s \) or \( r = s + 1 \).

DIMSIMs can be classified into four types according to the classification of GLMs introduced in [8]. Type 1 or type 2 methods are those with arbitrary \( a_{ij} \) and \( \lambda = 0 \) or \( \lambda \neq 0 \) and they are suited for non-stiff or stiff differential systems respectively in a sequential computing environment. Type 3 or type 4 methods are corresponding parallel versions requiring all non-diagonal elements in \( A \) be zero. Various DIMSIMs have been constructed with some specific stability properties. For type 1 and type 2 methods, it is usually desired that they have the inherited Runge-Kutta stability [11, 12].

Next we briefly review the construction of type 1 DIMSIMs with \( p = q = r = s \), \( U = I \), and \( V = ev^T \), where \( v'e = 1 \) since we will need to follow some procedures here when constructing IMEX-DIMSIM schemes. For details regarding construction of Type 2 methods, we refer readers to [10].

Imposing the appropriate stage order and order conditions and additional stability requirement results in large systems of polynomial equations for the remaining unknown parameters of the methods. If the order of the methods is less than 5, these systems can be generated and solved symbolic manipulation packages such as MATHEMATICA or MAPLE.

By design, the stability function \( p(w, z) \) has the form
\[
p(w, z) = w^{s-1}(w - R(z)),
\]
where \( R(z) \) is the stability function of RK method of order \( p = s \). It can be demonstrated that the stability function of DIMSIM of type 1 is a polynomial of the form
\[
p(w, z) = w^s - p_1(z)w^{s-1} + \ldots + (-1)^{s-1}p_{s-1}(z)w + (-1)^sp_s(z),
\]
where
\[
\begin{align*}
p_1(z) &= 1 + p_{11}z + p_{12}z^2 + \ldots + p_{1s}z^s, \\
p_2(z) &= p_{21}z + p_{22}z^2 + \ldots + p_{2s}z^s, \\
&\vdots \\
p_{s-1}(z) &= p_{s-1,s-2}z^{s-2} + p_{s-1,s-1}z^{s-1} + p_{s-1,s}z^s, \\
p_s(z) &= p_{s,s-1}z^{s-1} + p_{ss}z^s.
\end{align*}
\]
Note that the coefficients \( p_{ij} \) of the polynomials \( p_i(z) \) depend on \( a_{ij}, v_i \). This leads to the system of \((s - 1)(s + 2)/2\) nonlinear equations
\[
p_{kl} = 0, \ k = 2, 3, \ldots, s, \ l = k - 1, k, \ldots, s,
\]
with respect to \((s - 1)(s + 2)/2\) unknowns \( a_{ij} \) and \( v_i \).

It is proved in [10] that the DIMSIM \((A, U, B, V)\) has order \( p \) and stage order \( q \) equal to \( q = p = r = s \) if and only if
\[
B = B_0 - AB_1 - VB_2 + VA,
\]
where \( B_0, B_1, \) and \( B_2 \) are \( s \times s \) matrices with elements defined by
\[
\begin{align*}
\int_{0}^{1+c_i} \frac{\phi_j(x)dx}{\phi_j(c_j)}, \quad \int_{0}^{c_i} \frac{\phi_j(1+c_i)dx}{\phi_j(c_j)}, \quad \int_{c_i}^{c_j} \frac{\phi_j(x)dx}{\phi_j(c_j)},
\end{align*}
\]
and \( \phi_i(x) \) is evaluated by
\[
\phi_i(x) = \prod_{j=1, j\neq i}^{s} (x - c_j), \ i = 1, 2, \ldots, s.
\]
To generate dense output, we consider approximations for \( h^k y^k(t_0) \) of the form

\[
h^k y^k(t_0) \approx \sum_{i=0}^{s} \beta_{ki} f(Y_i) + \sum_{j=0}^{r} \gamma_{kj} y_j^{[n-1]}, \quad k = 0, 1, \ldots, r.
\]

It is shown in [8, 13] that these approximations are correct within \( O(h^{p+1}) \) if and only if

\[
[1, z, \ldots, z^p]^T \ e^z = zB e^z + \tilde{V} w(z) + O(z^{p+1})
\]

(12)

where \( \tilde{V} = [\beta_{ki}], \tilde{V} = [\gamma_{kj}], e^{cz} = [e^{cz_1}, \ldots, e^{cz_p}]^T \) and \( w(z) = \sum_{j=0}^{p} a_j z^j \). If we want to generate the solution at the last step, we can simply use the first equation in 12 when \( k = 0 \),

\[
y(t_0) \approx \sum_{i=0}^{s} \beta_{0i} f(Y_i) + \sum_{j=0}^{r} \gamma_{0j} y_j^{[n-1]}. \tag{13}
\]

3. Construction of IMEX-DIMSIM schemes

To derive an IMEX scheme, firstly we cast the system (1) as a partitioned system,

\[
\begin{align*}
\dot{x} &= \hat{f}(x, y), \tag{14a} \\
\dot{y} &= \hat{g}(x, y), \tag{14b}
\end{align*}
\]

with \( u = x + y, \hat{f} = f(x + y) \) and \( \hat{g} = g(x + y) \). Applying a type 1 DIMSIM with \( p = q = r = s \) and \( U = I \) to equation (14a) and a type 2 DIMSIM also with \( p = q = r = s \) and \( U = I \) to equation (14b) yields

\[
\begin{align*}
X_i &= h \sum_{j=1}^{s} \hat{a}_{ij} f(X_j + Y_j) + x_i^{[n-1]}, \quad i = 1, 2, \ldots, s, \tag{15a} \\
x_i^{[n]} &= h \sum_{j=1}^{s} \hat{b}_{ij} f(X_j + Y_j) + \hat{v}_{ij} x_j^{[n-1]}, \quad i = 1, 2, \ldots, r, \tag{15b}
\end{align*}
\]

and

\[
\begin{align*}
Y_i &= h \sum_{j=1}^{s} a_{ij} g(X_j + Y_j) + y_i^{[n-1]}, \quad i = 1, 2, \ldots, s, \tag{16a} \\
y_i^{[n]} &= h \sum_{j=1}^{s} b_{ij} g(X_j + Y_j) + v_{ij} y_j^{[n-1]}, \quad i = 1, 2, \ldots, r. \tag{16b}
\end{align*}
\]

Combine (15) and (16), and we have

\[
\begin{align*}
X_i + Y_i &= h \left( \sum_{j=1}^{s} \hat{a}_{ij} f(X_j + Y_j) + \sum_{j=1}^{s} a_{ij} g(X_j + Y_j) \right) + x_i^{[n-1]} + y_i^{[n-1]}, \quad i = 1, 2, \ldots, s, \tag{17a} \\
x_i^{[n]} + y_i^{[n]} &= h \left( \sum_{j=1}^{s} \hat{b}_{ij} f(X_j + Y_j) + \sum_{j=1}^{s} b_{ij} g(X_j + Y_j) \right) + \sum_{j=1}^{r} \hat{v}_{ij} x_j^{[n-1]} + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, \quad i = 1, 2, \ldots, r. \tag{17b}
\end{align*}
\]

If the type 2 DIMSIM (implicit) scheme shares the same abscissa vector \( c = [c_1, c_2, \ldots, c_s]^T \) and the same coefficient matrix \( V \) with the type 1 DIMSIM (explicit) scheme, \( X_i \) and \( Y_i \) can be added up at all internal stages. Then we
can define a new internal stage vector as $Z_i = X_i + Y_i$ and obtain the IMEX-DIMSIM (or diagonally-implicit-explicit multi-stage integration method)

$$Z_i = h \left( \sum_{j=1}^{s} \hat{a}_{ij} f(Z_j) + \sum_{j=1}^{s} a_{ij} g(Z_j) \right) + x_i^{[n-1]} + y_i^{[n-1]}, \quad i = 1, 2, \ldots, s, \quad (18a)$$

$$x_i^{[n]} + y_i^{[n]} = h \left( \sum_{j=1}^{s} \hat{b}_{ij} f(Z_j) + \sum_{j=1}^{r} b_{ij} g(Z_j) \right) + \sum_{j=1}^{r} v_{ij} \left( x_j^{[n]} + y_j^{[n]} \right), \quad i = 1, 2, \ldots, r, \quad (18b)$$

The derived IMEX-DIMSIM scheme can also be seen as a partitioned DIMSIM scheme. We note that for (18) $x_i$ and $y_i$ need not to be known individually once they are initialized before the first step. The combined quantity $x_i + y_i$ advances at each step as other normal DIMSIMs do.

According to the definition of DIMSIM, the initial values $x_i^{[0]}$ and $y_i^{[0]}$ should approximate linear combinations

$$\sum_{k=0}^{r} \hat{a}_{ik} h^k x^{(k)}(t_0) \quad \text{and} \quad \sum_{k=0}^{r} \alpha_{ik} h^k y^{(k)}(t_0)$$

respectively where

$$\hat{a}_0 = e, \quad \hat{a}_i = \frac{c_i^j}{i!} - \frac{\hat{A} c_i^{j-1}}{(i-1)!},$$

$$\alpha_0 = e, \quad \alpha_i = \frac{c_i^j}{i!} - \frac{A c_i^{j-1}}{(i-1)!}.$$ 

Thus

$$x_i^{[0]} + y_i^{[0]} = x(t_0) + y(t_0) + \hat{a}_{11} h x'(t_0) + \alpha_{11} h y'(t_0) + \sum_{k=2}^{r} \hat{a}_{ik} h^k x^{(k)}(t_0) + \sum_{k=2}^{r} \alpha_{ik} h^k y^{(k)}(t_0)$$

$$= u(t_0) + \hat{a}_{11} h f(u(t_0)) + \alpha_{11} h g(u(t_0)) + \sum_{k=2}^{r} \hat{a}_{ik} h^k x^{(k)}(t_0) + \sum_{k=2}^{r} \alpha_{ik} h^k y^{(k)}(t_0).$$

Evaluation of the first three terms is straightforward. But approximations of the other terms containing derivatives $x^{(k)}(t_0)$ and $y^{(k)}(t_0)$ (of order higher than 1) remain to be further addressed if their analytical expressions are hard to obtain.

To generate the solution at the last time step $u(t_F)$ using (13), we must put more constraints on the coefficients $\gamma$. For type 1 DIMSIMs, if we choose the abscissa vector to be $[0, e_2, \ldots, e_r]^T$, the first element of the vector $y$ is exactly the solution at the current step since $\alpha_{00} = 1$ and $\alpha_{0i} = 0$ for $i > 0$ according to order conditions. In this case, $\beta_0$ is equal to the first row of coefficient matrix $B$ while $\gamma_0$ to the first row of $V$. For type 2 DIMSIMs, there are usually some free parameters left in $B$ and $V$ after satisfying (10). These free parameters could be chosen in such a way that the type 1 DIMSIM shares the same coefficients $\gamma_0$ with the type 2 DIMSIM, and the difficulty of computing terms $x_i^{[n-1]}$ and $y_i^{[n-1]}$ individually can be avoided. We will illustrate this with specific examples in the following.

4. A two-stage, second-order pair with $p = q = r = s = 2$

A second-order accurate type 2 DIMSIM with exactly the same linear stability as the diagonally-implicit Runge-Kutta method is [9]

$$\begin{bmatrix}
\frac{2 - \sqrt{3}}{2} & 0 & 1 \\
\frac{2\sqrt{3} + 6}{3} & \frac{2 - \sqrt{3}}{2} & 0 \\
\frac{73 - 34 \sqrt{3}}{28} & \frac{4\sqrt{3} - 5}{4} & \frac{3 - \sqrt{3}}{2} \\
\frac{87 - 48 \sqrt{3}}{28} & \frac{1 - \sqrt{3}}{4} & \frac{3 - \sqrt{3}}{2}
\end{bmatrix}. $$
The corresponding type 1 DIMSIM should have the same $U$, $V$ and abscissa vector $[0, 1]$. Thus the only free parameter is $\hat{a}_{21}$ since $B$ is determined by $A$, $V$ and abscissa vector in (10) as required by the order condition.

The stability polynomial is

$$p(w, z) = w^2 - w \left[ 1 + \frac{6 - \sqrt{3}}{2} \hat{a}_{21} z + \frac{3 - \sqrt{3}}{4} \hat{a}_{21}^2 z^2 \right] + \left( \frac{\sqrt{3} - 3}{2} \hat{a}_{21} - \sqrt{2} + \frac{3}{4} \right) z + \left( \frac{\sqrt{3} - 3}{2} (\hat{a}_{21} - 2) \right) z^2. \quad (19)$$

Inheriting Runge-Kutta stability requires the coefficients $p_{21} = 0$ and $p_{22} = 0$. However these two equations cannot be fulfilled at the same time. We choose to meet the condition $p_{22} = 0$, which yields $\hat{a}_{21} = 2$, just to make its stability properties 'close' to Runge-Kutta methods. The resulting type 1 DIMSIM is

$$\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} \frac{3 \sqrt{3} - 1}{4} & \frac{3 - \sqrt{3}}{4} \\ \frac{3 \sqrt{3} - 3}{4} & \frac{3 - \sqrt{3}}{4} \end{bmatrix}, \begin{bmatrix} \frac{3 - \sqrt{3}}{2} z \end{bmatrix}. \quad (20)$$

Actually in the combination we can make only one of the two methods to inherit RK or SDIRK stability because there are not enough free parameters to fit the stability requirement. And it is much more important to impose good stability on the type 2 DIMSIM dealing with stiff terms.

According to the equation (13), the final solution can be approximated within the desired order by applying the following formula

$$u(t_n) \approx \hat{\beta}_{01} f(Z_1) + \hat{\beta}_{02} f(Z_2) + \beta_{01} g(Z_1) + \beta_{02} g(Z_2) + \hat{\gamma}_{01} x_1^{[n-1]} + \hat{\gamma}_{02} x_2^{[n-1]} + \gamma_{01} y_1^{[n-1]} + \gamma_{02} y_2^{[n-1]} . \quad (21)$$

Since $c_0 = 0$, coefficients for the type 1 DIMSIM equal to the first rows of $B$ and $V$

$$\hat{\beta}_{01} = \hat{b}_{11} = \frac{3 \sqrt{3} - 1}{4}, \quad \hat{\beta}_{02} = \hat{b}_{12} = \frac{3 - \sqrt{3}}{4}, \quad \hat{\gamma}_{01} = v_{11} = \frac{3 - \sqrt{3}}{2}, \quad \hat{\gamma}_{02} = v_{12} = \frac{\sqrt{3} - 1}{2} .$$

Solving the order condition (12) gives

$$\beta_{01} = \frac{73 - 34 \sqrt{3}}{28} + \frac{43 - 31 \sqrt{5}}{28} g, \quad \beta_{02} = \frac{-1 + 2 \sqrt{2}}{4} + \frac{-4 + 3 \sqrt{5}}{4} g, \quad \gamma_{01} = \frac{3 - \sqrt{3}}{2} + \frac{2 - \sqrt{2}}{2} g, \quad \gamma_{02} = \frac{\sqrt{3} - 1}{2} + \frac{\sqrt{2} - 2}{2} g .$$

The choice of the free parameter $g = 0$ can make $\gamma_{01} = \hat{\gamma}_{01}$ and $\gamma_{02} = \hat{\gamma}_{02}$, leading to

$$u(t_n) \approx \hat{\beta}_{01} f(Z_1) + \hat{\beta}_{02} f(Z_2) + \beta_{01} g(Z_1) + \beta_{02} g(Z_2) + \hat{\gamma}_{01} x_1^{[n-1]} + \gamma_{01} y_1^{[n-1]} + \hat{\gamma}_{02} x_2^{[n-1]} + \gamma_{02} y_2^{[n-1]} , \quad (21)$$

where the quantity $x^{[n-1]} + y^{[n-1]}$ can be obtained directly by the time stepping scheme (18) we developed.

5. Numerical results

To confirm the theoretical order of our derived IMEX-DIMSIM scheme, we conduct some tests on the nonlinear van der Pol equation on the time interval $[0, 0.5]$

$$\begin{bmatrix} y' \\ z' \end{bmatrix} = f(y, z) + g(y, z) = \begin{bmatrix} z \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ (1 - y^2)z - y \end{bmatrix} / \varepsilon \quad (22)$$

with initial values

$$y(0) = 2, \quad z(0) = \frac{2}{3} + \frac{10}{81} \varepsilon - \frac{292}{2187} \varepsilon^2 - \frac{1814}{19683} \varepsilon^3 + O(\varepsilon^4). \quad (23)$$
We choose the stiffness parameter $\varepsilon = 0.001$ to make the problem stiff. In our experiment, step sizes $h_0$, $h_0/2$, $h_0/4$ and $h_0/8$ with $h_0 = 0.01$ were used. And the initialization of the quantity $x_i + y_i$ was done by computing the analytic derivatives by hand. For reference solutions, we used MATLABs ode15s routine with very small tolerances $atol = rtol = 2.22045e-14$.

We compare the diagonally-implicit-explicit multi-stage integration method with implicit-explicit midpoint method, which is a second-order IMEX Runge-Kutta method considered in [4]. In Figure 1 we have plotted the global error, measured in the $L_2$ norm, against step size $h$. Both methods show an observed order of 2, which agrees exactly with the predicted order of accuracy. For the same step size, our method is more accurate than the second order IMEX Runge-Kutta method.

6. Conclusions and future work

In this paper we have introduced a new class of IMEX schemes based on pairs of diagonally-implicit multi-stage integration methods to fill in the gap where no existing IMEX-GLMs have been developed. We show the construction of a second-order diagonally-implicit-explicit multi-stage integration method and provide the corresponding coefficients. Numerical experiment on the stiff van del Pol equation confirms that their observed orders match the theoretical results.

In the future work we will develop IMEX-DIMSIMs of higher orders and investigate their properties such as stability and convergence, as well as their advantages compared to other existing IMEX schemes. There are also some implementation issues that deserve further exploration.

Acknowledgement

This work has been supported in part by NSF through awards NSF OCI-8670904397, NSF CCF-0916493, NSF DMS-0915047, NSF CMMI-1130667, and by the Computational Science Laboratory at Virginia Tech.

References

[1] U. M. Ascher, S. J. Ruuth, B. T. R. Wetton, Implicit-explicit methods for time-dependent partial differential equations, SIAM Journal on Numerical Analysis 32 (3) (1995) pp. 797–823.
[2] J. Frank, W. Hundsdorfer, J. Verwer, On the stability of implicit-explicit linear multistep methods, Applied Numerical Mathematics 25 (23) (1997) 193 – 205, special Issue on Time Integration. doi:10.1016/S0168-9274(97)00059-7.
[3] W. Hundsdorfer, S. J. Ruuth, IMEX extensions Of linear multistep methods with general monotonicity, Journal of Computational Physics 225 (2) (2007) 2016 – 2042.
[4] U. M. Ascher, S. J. Ruuth, R. J. Spiteri, Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations, Appl. Numer. Math 25 (1997) 151–167.
[5] S. Boscarino, On an accurate third order implicit-explicit Runge–Kutta method for stiff problems, Appl. Numer. Math. 59 (2009) 1515–1528. doi:10.1016/j.apnum.2008.10.003.
[6] L. Pareschi, G. Russo, Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation, Journal of Scientific Computing 3 (2000) 269–287.
[7] J. Verwer, B. Sommeijer, W. Hundsdorfer, RKC time-stepping for advection-diffusion-reaction problems, Journal of Computational Physics 201 (1) (2004) 61–79. doi:10.1016/j.jcp.2004.05.002.
[8] J. C. Butcher, Z. Jackiewicz, Diagonally implicit general linear methods for ordinary differential equations, BIT Numerical Mathematics 33 (1993) 452–472. 10.1007/BF01990528.
[9] J.C., Butcher, Diagonally-implicit multi-stage integration methods, Applied Numerical Mathematics 11 (5) (1993) 347 – 363. doi:10.1016/0168-9274(93)90059-Z.
[10] Z. Jackiewicz, General linear methods for ODE, John Wiley & Sons, Inc., 2009. doi:10.1002/9780470522165.fmatter.
[11] W. M. Wright, General linear methods with inherent Runge-Kutta stability, Ph.D. thesis, The University of Auckland (2002).
[12] J. Butcher, W. Wright, The construction of practical general linear methods, BIT Numerical Mathematics 43 (2003) 695–721, 10.1023/B:BITN.0000009952.71388.23.
[13] J. C. Butcher, Z. Jackiewicz, Implementation of diagonally implicit multistage integration methods for ordinary differential equations, SIAM Journal on Numerical Analysis 34 (6) (1997) pp. 2119–2141.