2019-07-01

Runge–Kutta–Gegenbauer Explicit Methods for Advection-Diffusion Problems

Stephen O'Sullivan

Technological University Dublin, stephen.osullivan@tudublin.ie

Recommended Citation

O'Sullivan, S. (2019) Runge–Kutta–Gegenbauer explicit methods for advection-diffusion problems, *Journal of Computational Physics,* Vol. 388, 1 July 2019, Pages 209-223, doi.org/10.1016/j.jcp.2019.03.001

Runge–Kutta–Gegenbauer explicit methods for advection-diffusion problems

Stephen O'Sullivan*[1]

[1]School of Mathematical Sciences, Technological University Dublin, Kevin Street, Dublin 8, Ireland

April 18, 2019

Abstract

In this paper, Runge–Kutta–Gegenbauer (RKG) stability polynomials of arbitrarily high order of accuracy are introduced in closed form. The stability domain of RKG polynomials extends in the the real direction with the square of polynomial degree, and in the imaginary direction as an increasing function of Gegenbauer parameter. Consequently, the polynomials are naturally suited to the construction of high order stabilized Runge–Kutta (SRK) explicit methods for systems of PDEs of mixed hyperbolic-parabolic type.

We present SRK methods composed of L ordered forward Euler stages, with complex-valued stepsizes derived from the roots of RKG stability polynomials of degree L. Internal stability is maintained at large stage number through an ordering algorithm which limits internal amplification factors to $10L^2$. Test results for mildly stiff nonlinear advection-diffusion-reaction problems with moderate ($\lesssim 1$) mesh Péclet numbers are provided at second, fourth, and sixth orders, with nonlinear reaction terms treated by complex splitting techniques above second order.

Keywords: Stiff equations, Stability and convergence of numerical methods, Method of lines
2010 MSC: 65L04 , 65L20 , 65M20

1 Introduction

Stabilized Runge–Kutta (SRK) explicit methods are particularly well suited to solving mildly stiff systems of ODEs arising from the discretization of parabolic PDEs due to their extended stability domains along the negative real axis. In this work, we will consider an extension of this class of methods to systems of ODEs derived from PDEs of mixed hyperbolic-parabolic type.

The canonical m-dimensional scalar advection-diffusion equation for a quantity w is given by

$$w_t + \sum_{k=1}^{m} a_k w_{x_k} = d \sum_{k=1}^{m} w_{x_k x_k}, \tag{1}$$

*Email address: stephen.osullivan@dit.ie

where a_k is the advection coefficient in the k-th direction and d is the diffusion coefficient. Assuming a spatial mesh of uniform spacing h_k in the k-th direction, spatial discretization leads to a system of ODEs via the method of lines which may be written in the form

$$w' = f(t, w). \tag{2}$$

Given an initial state w_0, subsequent application of a numerical integration method over n steps then yields an approximate solution w^n at time t^n.

Under the description given above, the mesh Péclet number associated with the k-th dimension, $P_k = |a_k|h_k/d$, describes the relative significance of the hyperbolic to the parabolic parts of eq. (1). In particular, the domain of the eigenvalues from a von Neumann stability analysis increases in the imaginary direction with P_k. A well constructed numerical method must therefore capture this stability domain to avoid unbounded growth of errors.

SRK methods based on Chebyshev polynomials (eg. RKC [25], DUMKA [20], ROCK [2], and FRKC [23]) are suitable for problems in the limit of vanishing Péclet numbers. The stability domains associated with these polynomials are extended along the real axis, however, in unmodified form, they also possess internal points where the domains become vanishingly narrow. In practice, damping procedures have been used to introduce a finite imaginary extent at these points of marginal stability so as to mitigate against instability arising through truncation errors. A damping process has been exploited in [27] to extend applicability of RKC methods to problems with moderate ($\lesssim 1$) mesh Péclet numbers.[1] The Extrapolated Stabilized Explicit RungeKutta (ESERK) methods of [19] are derived from Richardson extrapolation techniques and demonstrate finite extent stability domains in the imaginary sense through damping. Finally, we remark that methods based on Legendre polynomials have also been proposed [21, 22] which do not suffer from marginally stable internal points and therefore do not require damping for problems with very small ($\ll 1$) mesh Péclet numbers.

In this work, we demonstrate that the necessity for modification of SRK methods through a damping procedure for problems with mild-to-moderate advection may be avoided by appealing to the properties of the general class of Gegenbauer polynomials. We seek a closed-form prescription for arbitrarily high order Runge–Kutta–Gegenbauer (RKG) stability polynomials which natively generate stability domains with imaginary extent determined by the Gegenbauer parameter.

The paper is organized as follows. In section 2, the analytic form of the class of RKG stability polynomials is presented and the construction of stable time-marching explicit methods based on the roots of these polynomials is outlined. In section 3, numerical tests are presented confirming the order and efficiency properties of RKG methods. Conclusions are presented in section 4.

2 Runge-Kutta-Gegenbauer methods

2.1 Runge-Kutta-Gegenbauer stability polynomials

By appending t to the vector of dependent variables, eq. (2) may be written in autonomous form,

$$w' = f(w). \tag{3}$$

We consider advancing the approximate solution explicitly over a single timestep T from w^n, at time level n, to w^{n+1}, at time level $n+1$. In the following discussion, an order N SRK method is implemented over L stages spanning an aggregate time T. For clarity of notation, we omit the time level indexing and denote the $L+1$ internal stage

[1] The authors also consider large (> 1) mesh Péclet numbers, however, the benefits of using SRK methods in terms of efficiency are largely lost.

states W^l ($l = 0, \cdots, L$), with $W^0 = w^n$, and $W^L = w^{n+1}$. The SRK integration then takes the form

$$W^L = W^0 + T\sum_{l=1}^{L} a_l f(W^{l-1}), \tag{4}$$

where the timestep related to each stage is given by $\tau_l = a_l T$.

When applied to the scalar Dahlquist test equation

$$W'(t) = \lambda W(t), \quad W(0) = 1, \tag{5}$$

the RKG method of rank N, degree $L = MN$, and Gegenbauer parameter ν, may be written as

$$W^L = R_M^{\nu N}(z), \tag{6}$$

where $z = T\lambda$. The associated RKG stability function,

$$R_M^{\nu N}(z) = G_M^{\nu N}\left(1 + \frac{2z}{\beta_M^{\nu N}}\right), \tag{7}$$

is obtained from the shifted RKG polynomial

$$G_M^{\nu N}(z) = d_0^{\nu N} + 2\sum_{k=1}^{N} d_k^{\nu N} C_{kM}^{\nu}(z), \tag{8}$$

where C_{kM}^{ν} denotes the the Gegenbauer polynomial of degree kM and Gegenbauer parameter ν. [2] We note that since any perturbation will be amplified by $R_M^{\nu N}(z)$ over a timestep, the method's stability domain is defined by $\{z \in \mathbb{C} : |R_M^{\nu N}(z)| \leq 1\}$.

Linear order conditions are obtained by requiring that the first N terms of the Taylor series for the exact solution of eq. (5) and $R_M^{\nu N}(z)$, coincide. Hence, the order coefficients, $d_k^{\nu N}$, are determined by

$$R_M^{\nu N\,(n)}(0) = 1, \quad n = 1, \ldots, N, \tag{9}$$

where a superscript (n) indicates the nth derivative is taken. The order conditions are met by solving an N-dimensional linear system

$$\begin{bmatrix} C_M^{\nu\,(1)}(1) & \ldots & C_{NM}^{\nu\,(1)}(1) \\ \vdots & \ddots & \vdots \\ C_M^{\nu\,(N)}(1) & \ldots & C_{NM}^{\nu\,(N)}(1) \end{bmatrix} \begin{bmatrix} d_1^{\nu N} \\ \vdots \\ d_N^{\nu N} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} (\beta_M^{\nu N}/2)^1 \\ \vdots \\ (\beta_M^{\nu N}/2)^N \end{bmatrix}, \tag{10}$$

coupled with a zeroth order condition

$$d_0^{\nu N} = 1 - 2\sum_{k=1}^{N} d_k^{\nu N} C_{kM}^{\nu}(1). \tag{11}$$

Once the value of $\beta_M^{\nu N}$ has been set, the above equations are sufficient to determine $R_M^{\nu N}$: for odd M, optimal values of $\beta_M^{\nu N}$ are found by solving $G_M^{\nu N}(-1) = (-1)^N$ iteratively; for even M, rational interpolation/extrapolation is used.

In this work, we consider RKG polynomials of order $N = 1, \ldots, 8$ and adopt geometric sequences of values for ν, from 0 to $2N$, given by $\nu = 2^{i/2}N/128$, for $i = 0, \ldots, 16$, and also include the reference value $\nu = 0$. In considering the gain in efficiency achievable by methods constructed from the RKG polynomial, it is useful to note that M aggregated steps of a standard RK methods at order N will have a stability polynomial of degree L and a domain with a real extent of approximately $M\beta_1^{\nu N}$. From fig. 1, it may be seen that the real extent, $\beta_M^{\nu N}$, of an RKG polynomial

[2] We remark that for $\nu = 0$ and $\nu = 1/2$ the components are Chebyshev and Legendre polynomials respectively.

of the same degree is characteristically longer by a factor of between M and $1.5M$ for $\nu = 0$, falling to $\lesssim 0.5M$ with increasing ν. This factor therefore represents a significant potential gain in efficiency with respect to standard explicit RK methods. A count of the number of evaluations, f_{count}, for the function f in eq. (4), may be used as a measure of the work required to carry out an integration. We note here that the optimal efficiency for SRK methods follows $f_{\text{count}} \propto (\text{err})^{-1/2N}$ (where err is a measure of the error in the solution).

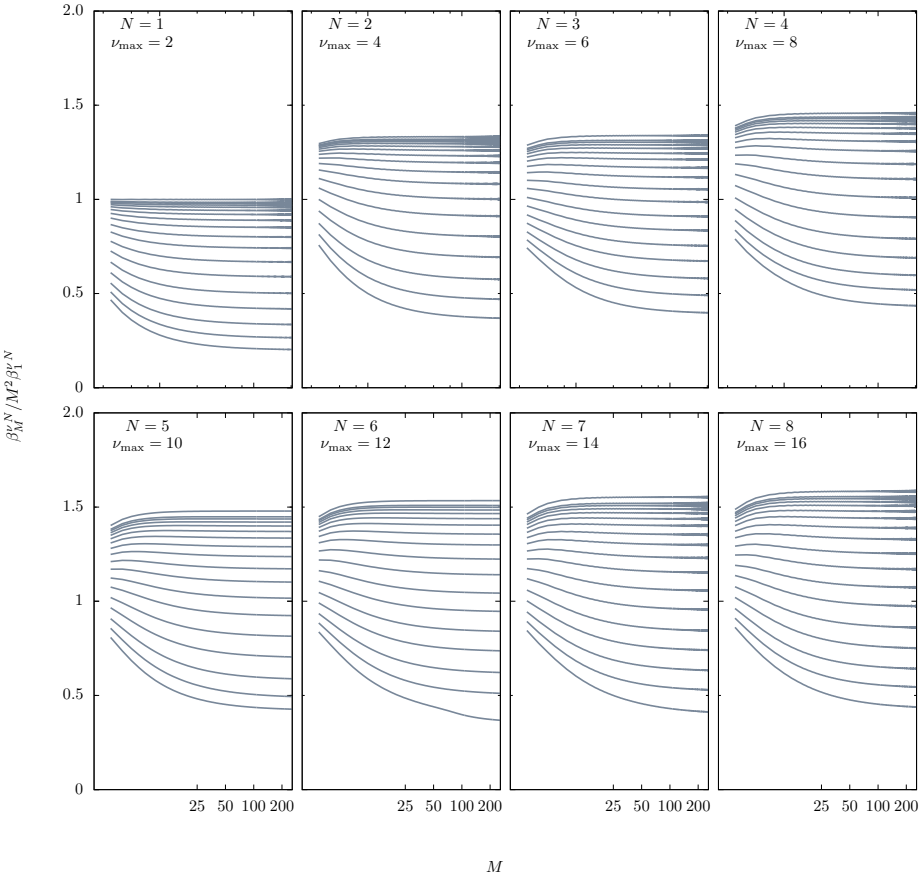

Figure 1: Stability domain extent, for $|R_M^{\nu N}| = 1$, along the real axis normalized to $M^2\beta_1^{\nu N}$ as a function of M for orders $N = 1, \ldots, 8$. The Gegenbauer parameter ν ranges from 0 to $2N$ (from top to bottom) with $\nu = 0$ and $\nu = 2^{i/2}N/128$, for $i = 0, \ldots, 16$.

2.2 Factorized Runge–Kutta–Gegenbauer methods

The Factorized Runge–Kutta–Gegenbauer (FRKG) coefficients of order N corresponding to eq. (4) are defined according to

$$a_{Ml}^{\nu N} = \frac{2}{\beta_M^{\nu N}} \frac{1}{1 - \zeta_{Ml}^{\nu N}}, \tag{12}$$

where the roots $\zeta_{Ml}^{\nu N}$ of the RKG polynomial $G_M^{\nu N}(z)$ are determined numerically.[3] The maximum stable stepsize for the derived method is $T_{\max} = \beta_M^{\nu N}/|\lambda|_{\max}$, where the values of λ are the negative-definite eigenvalues of the Jacobian associated with eq. (3).

Over the extent of the stability domain on the real axis, with $\nu > 0$, we remark that $|R_M^{\nu N(n)}| < 1$. In the limiting case of $\nu = 0$, the components of $R_M^{\nu N}(z)$ are shifted Chebyshev polynomials and $M - 1$ marginally stable internal points exist for which $|R_M^{\nu N(n)}| = 1$. For problems with formally real eigenvalues under von Neumann stability analysis, truncation errors may give rise to eigenvalues with small imaginary parts thereby introducing some susceptibility to instability. In this work, we therefore choose a minimum value of $\nu = N/128$ for all test cases so as to avoid this phenomenon.

For various standard discretization schemes, [29] determined geometric shapes within which the eigenvalues derived from a von Neumann stability analysis will be contained. While [27] adopt an oval approximation, we find that better results are found for large M using the ellipse approximation

$$\left(\frac{x}{\beta_M^{\nu N}/2} + 1\right)^2 + \left(\frac{y}{\alpha_M^{\nu N}}\right)^2 = 1, \tag{13}$$

where the centre point is $(-\beta_M^{\nu N}/2,\, 0)$, and the major and minor half-axes are $\beta_M^{\nu N}/2$ and $\alpha_M^{\nu N}$ respectively.

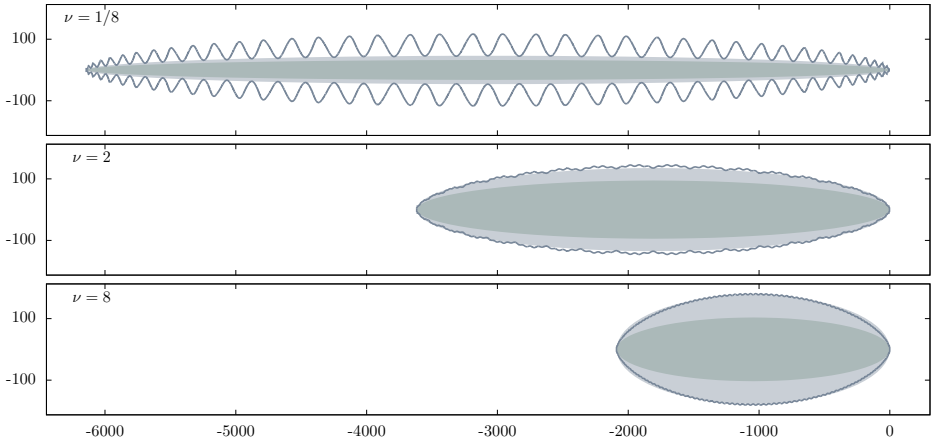

Figure 2: Stability domains for fourth order RKG stability polynomials with $M = 40$ and $\nu = 1/8,\, 2,\, 8$. The shaded regions indicate the strict and approximate fitted ellipses, corresponding to the semi-minor axes $\alpha_{s\,M}^{\nu N}$ and $\alpha_{a\,M}^{\nu N}$ respectively. (The cases shown are characteristic of the methods used in obtaining solutions with $err \gtrsim 10^{-5}$ for the Brusselator problem with advection presented in section 3 with weak, mild, and moderate advection, from top to bottom.)

Under κ-scheme discretizations [26], the stability condition derived by [29] for the ellipse approximation is given by

$$T \leq \min\left(\psi_1 \beta_M^{\nu N},\, \psi_2 \frac{(\alpha_M^{\nu N})^2}{\beta_M^{\nu N}}\right), \tag{14}$$

[3]Root-finding is handled with the MPSolve [3] package. Multiple-precision calculations of polynomial and method coefficients are carried out using the GMP [9] and MPFR [7] libraries.

where

$$\psi_1 = \frac{1}{2d\sum h_k^{-2}(2+(1-\kappa)P_k)}, \quad \psi_2 = \frac{4d}{(2-\kappa)^2\sum a_k^2}. \tag{15}$$

We identify an ellipse with semi-minor axis, $\alpha_{s\,M}^{\nu\,N}$, which is fit strictly to the interior of the contour $|R_M^{\nu\,N}(z)| = 1$. Typically, this strict ellipse turns out to be excessively conservative and we therefore also make use of an approximate ellipse with semi-minor axis $\alpha_{a\,M}^{\nu\,N}$. The value of $\alpha_{a\,M}^{\nu\,N}$ is set by solving $|R_M^{\nu\,N}(\mathrm{i}\alpha_{a\,M}^{\nu\,N})| = 1$ for $M \bmod 4 = 0$, and logarithmically interpolating for the intermediate cases. [4] Except for $M \leq 4$, the approximate ellipse semi-minor axis $\alpha_{a\,M}^{\nu\,N}$ is larger than the strict value $\alpha_{s\,M}^{\nu\,N}$.

Examples of the stability domains for various RKG stability polynomials are shown in fig. 2, superimposed with the exact and approximate ellipses corresponding to $\alpha_{s\,M}^{\nu\,N}$ and $\alpha_{a\,M}^{\nu\,N}$ respectively. Evidently, the approximate ellipse is well captured within the true stability domain, only overshooting marginally as ν becomes large. Furthermore, as the amplitude of the ripples in the stability domain boundary decreases with increasing ν, the approximate ellipse becomes an increasingly precise global approximation.

Algorithm 1 Selection of M and ν for timestep T ($\nu = \nu(i)$, increasing with i)

```
Initialize M = 1, i = 1
loop
  if ψ1 β_M^{νN} ≤ ψ2 (α_M^{νN})^2 / β_M^{νN} then
    if β_M^{νN} / |λ|_max < T then
      Select M, ν = ν(i)
    else
      Increment M
    end if
  else
    Increment i
  end if
end loop
```

In cases where a fixed timestep is assigned, we find experimentally that $\alpha_M^{\nu\,N} = \max(\alpha_{s\,M}^{\nu\,N}, (\alpha_{s\,M}^{\nu\,N} + \alpha_{a\,M}^{\nu\,N})/2)$ is an effective choice for eq. (14). Alternatively, when a timestep controller is employed (discussed further in section 2.5), integration is less sensitive to capturing the stability domain completely and $\alpha_M^{\nu\,N} = \max(\alpha_{s\,M}^{\nu\,N}, \alpha_{a\,M}^{\nu\,N})$ proves to be a suitable choice. (The max function evaluates to $\alpha_{s\,M}^{\nu\,N}$ only in a limited number of cases for $M \leq 4$.) As illustrated in fig. 3, the value of $(\alpha_M^{\nu\,N})^2/\beta_M^{\nu\,N}$ varies approximately linearly with ν over two orders of magnitude (from ~ 0.1 to ~ 10) for $0 \lesssim \nu \leq 2N$, and also demonstrates a general upward trend with increasing order N. The integration method is chosen according to algorithm 1 by selecting from the minimum available values of M and ν for which $\psi_1\beta_M^{\nu\,N} \leq \psi_2(\alpha_M^{\nu\,N})^2/\beta_M^{\nu\,N}$.

2.3 Internal stability

Internal instability arises when the product of any internal sequence of stages generates large values which drown out available numerical precision. Following an idea of Lebedev [16, 15], but using the more effective method presented in [23], and described here in algorithm 2, we order the stages in the L-tuple $[a_{M\,l}^{\nu\,N}]$ to approximately minimize

[4] Interpolation yields more conservative values than solving the given equation for $M \bmod 4 \neq 0$.

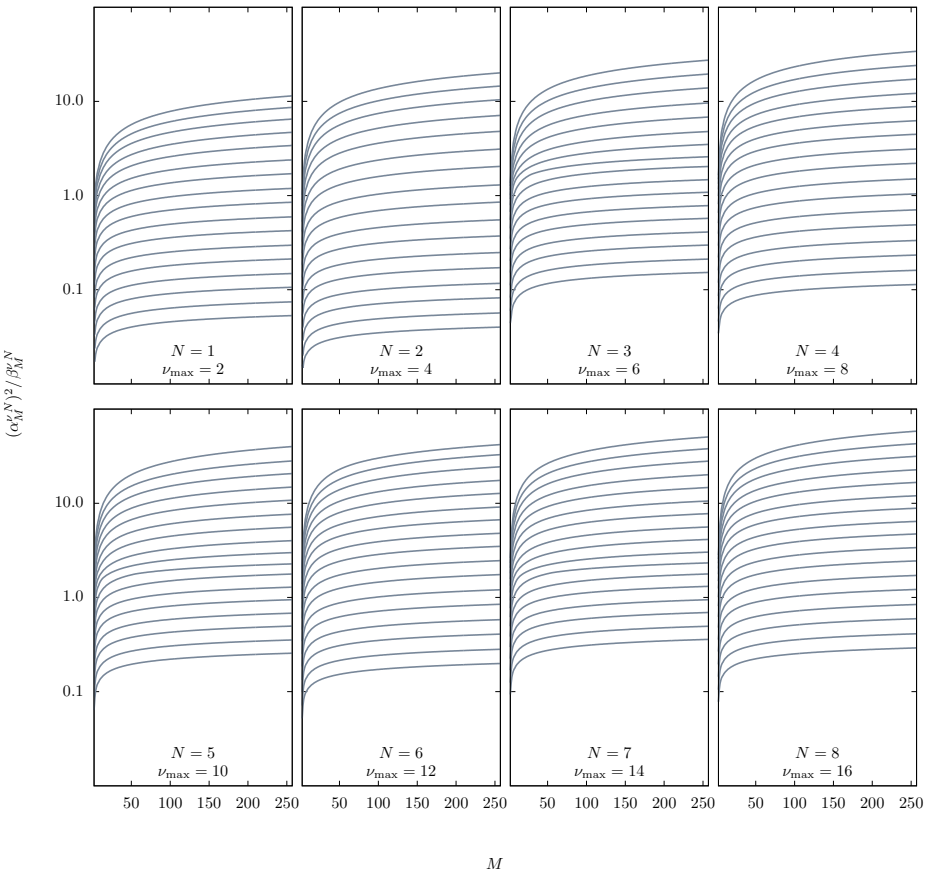

Figure 3: The parameter $(\alpha_M^{\nu N})^2/\beta_M^{\nu N}$ governing the stable timestep limit for the ellipse approximation to the stability domain as a function of M for orders $N = 1, \ldots, 8$ with $\nu = 2^{i/2}N/128$, for $i = 0, \ldots, 16$ (bottom to top). As shown, $(\alpha_M^{\nu N})^2/\beta_M^{\nu N}$ is approximately linear as a function of the Gegenbauer parameter ν.

the maximum realized internal amplification factor

$$\mathcal{Q} = \max(\mathcal{Q}_{j,\,k}(x)), \quad 1 \le j,\, k \le L, \quad x \in [-\beta_M^{\nu N},\, 0], \tag{16}$$

where

$$\mathcal{Q}_{j,\,k}(x) = \prod_{l=j}^{k} |1 + a_{Ml}^{\nu N} x|. \tag{17}$$

This is a rapid calculation: trialing each swap consists of multiplying the pre-existing forward amplification factor by $v_{j,\,k}$, and dividing the reverse factor by the same quantity. Adopting a slightly reduced stability range, $[(1 - 10^{-3}n)\beta_M^{\nu N},\, 0]$ ($n \in \mathbb{N}$), aids in meeting the imposed upper bound of $10L^2$ (particularly for small values of ν and N, where the mean value of n reaches almost 3). Preserving 7 digits for precision, a method consisting of $\sim 10^4$ stages is therefore theoretically viable in a numerical integration carried out at double (16 digit) precision - well beyond the requirements of practical calculations.

The maximum realized internal amplification factor $\mathcal{Q}$ is illustrated for $\nu = 0$ and $\nu = 2N$ over orders $N = 1, \ldots, 8$ in fig. 4. The imposed bound $10L^2$ is observed in all cases, with $\mathcal{Q}$ falling away from this limit as ν increases.[5]

Algorithm 2 Internal stability ordering of L-tuple $[a_{Ml}^{\nu N}]$

Shuffle $[a_{Ml}^{\nu N}]$
Initialize $n = 1$
repeat
  L points x_k uniformly on $[(1 - 10^{-3}n)\beta_M^{\nu N},\, 0]$, with $v_{j,\,k} = |1 + a_j x_k|$
  for $l = 1$ **to** $l = L$ **do**
    for $m = l$ **to** $m = L$ **do**
      Swap positions of $a_{Ml}^{\nu N}$ and $a_{Mm}^{\nu N}$
      if $\left\| \max\left(\prod_{j=1}^{l} v_{j,\,k},\, \prod_{j=l+1}^{L} v_{j,\,k} \right) \right\|_1$ is not new minimum over m **then**
        Revert positions of swapped coefficients $a_{Ml}^{\nu N}$ and $a_{Mm}^{\nu N}$
      end if
    end for
  end for
  Increment n
until $\mathcal{Q} < 10L^2$ {confirmed over $10L$ uniformly spaced points}

2.4 Complex splitting

Up to second order, FRKG methods are suitable without further consideration for nonlinear problems (since all order conditions are linear). At higher orders, nonlinear order conditions are present which require additional treatment. One approach to ensuring these nonlinear conditions are met is the composition of RK methods [5, 11], which offers a means of combining a linear SRK method with finishing stages derived from a nonlinear RK method [20, 2]. While composition methods are elegant, our studies and [14] suggest that poor error propagation properties over the finishing stages gives rise to erratic behaviour. Furthermore, the latter authors assert that composition methods are prone to order reduction whereby temporal accuracy is lost when spatial mesh refinement occurs.

[5] The ordered coefficients from first to eighth order, with $M = 1, \ldots, 257$, are provided as Electronic Supplementary Material.

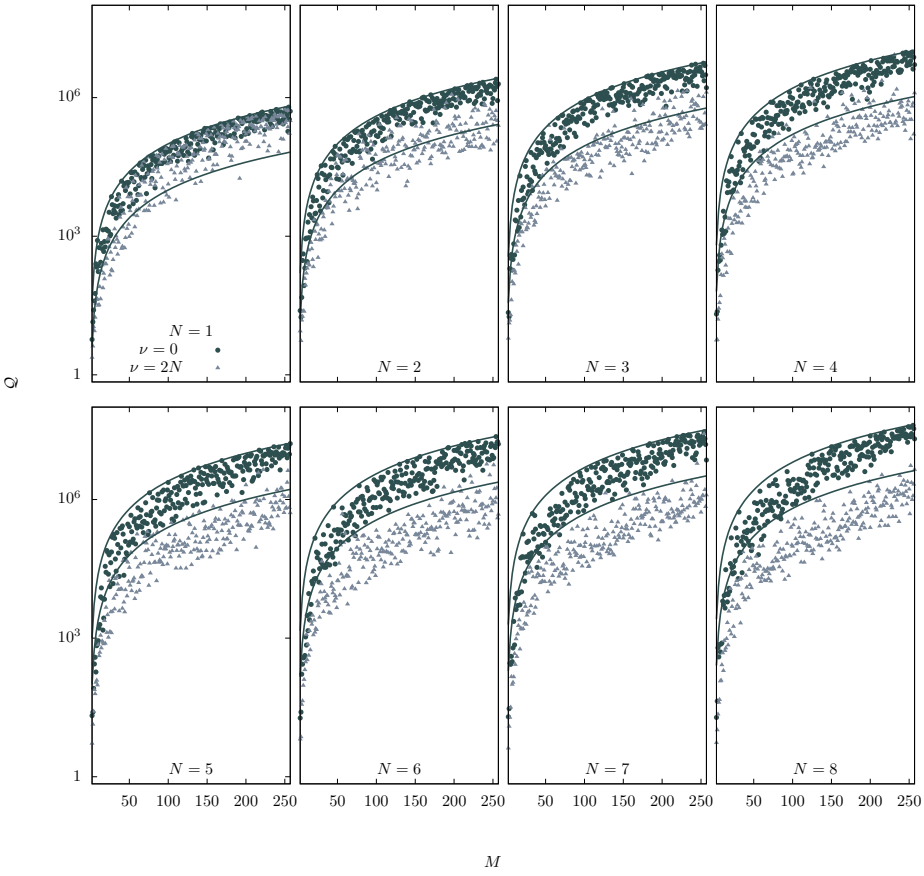

Figure 4: The maximum realized internal amplification factor $\mathcal{Q}$ as a function of M for FRKG integration methods over orders $N = 1, \ldots, 8$. The number of internal stages is given by $L = MN$. Cases for $\nu = 0$ (filled circles) and $\nu = 2N$ (filled triangles) are shown with guidelines at L^2 and $10L^2$.

An alternative approach for a linear evolution equation of the form $w' = (A + B)w$, is to employ a high order approximation of the operator $\mathrm{e}^{T(A+B)}$ in terms of complex valued coefficients [6, 13]. In this work, we adopt splitting of the form

$$w^{n+1} = \mathrm{e}^{T_{k_J} B} \mathrm{e}^{T_{k_{J-1}} A} \cdots \mathrm{e}^{T_{k_3} B} \mathrm{e}^{T_{k_2} A} \mathrm{e}^{T_{k_1} B} w^n, \tag{18}$$

where $T_k \in \mathrm{Re}^+$ for even k, corresponding to the partial timesteps for the operator A, and $T_k \in \mathrm{Im}^+$ otherwise [6, 4].

The procedure extends immediately to the case $w' = Aw + f_B(w)$, with B replaced by a nonlinear operator. The factors $\mathrm{e}^{T_k B}$ are replaced by appropriate approximations to the true flows given by $f_B(w)$ over the the intervals T_k.

Under Dirichlet or Neumann boundary conditions, splitting methods are known to be prone to an order reduction phenomenon. However, the effect appears to remain confined to the neighbourhood of the boundaries such that the order may be fully recovered on the interior of the domain [13, 18].

2.5 Stepsize control

For reliable numerical integration, an appropriate control procedure based on local error estimation is required to manage stepsize selection.

Starting from w^n, the approximate solutions w^{n+1} and $\overline{w}^{n+1}$ are obtained at order N, and at some lower order of accuracy $\overline{N}$, respectively. A measure of the error over the step is then given by

$$err^{n+1} = \left\| \frac{\overline{w}^{n+1} - w^{n+1}}{wt} \right\|_2, \tag{19}$$

where

$$wt = atol + \max(|w^n|, |w^{n+1}|) \times rtol, \tag{20}$$

with *atol* and *rtol* being tuning parameters for the absolute and relative errors respectively.

If $err^{n+1} > 1$, the step is rejected and w^{n+1} is recalculated with a revised timestep T_{new}. Otherwise, the solution is accepted and a trial solution w^{n+2} is determined over T_{new}. In order to prescribe T_{new}, it is observed that the error behavior for the FRKG methods follows $err \approx CT^{(N+1)/(N-\overline{N})}$. Hence, the constant C may be specified in terms of the error measure, and the revised stepsize may be estimated. For unsplit problems (when the previous step has been accepted) this is done by means of the the predictive controller [28, 10, 12] given by

$$T_{\mathrm{new}}^{n+1} = safe T^n \left(\frac{1}{err^{n+1}} \right)^{\frac{N-\overline{N}}{N+1}} \left(\frac{T^n}{T^{n-1}} \right) \left(\frac{err^n}{err^{n+1}} \right)^{\frac{N-\overline{N}}{N+1}}. \tag{21}$$

In all other cases, the non-predictive value obtained by deleting the third term in parentheses in eq. (21) is used. In practice, we choose $\overline{N} = 1$ for second order integrations, and $\overline{N} = 2$ at higher orders where splitting is applied. Additionally, we set the safety factor *safe* to 0.8 (after the initial step) and allow the revised timestep T_{new} to vary by at most a factor of two with respect to the previously trialed value.

For the initial step, following [25], the error is estimated over a trial step $T_{\mathrm{trial}} = 1/|\lambda|_{\max}$ by comparing forward Euler steps using function evaluations calculated at $t = 0$ and $t = T_{\mathrm{trial}}$. The initial step is then assigned using $T = safe \times T_{\mathrm{trial}}/\sqrt{err}$, with $safe = 0.1$.

2.6 Convex Monotone Property

Methods based on Chebyshev polynomials applied to pure diffusion problems with spatially varying diffusion coefficients may give rise to a staircase profile in the solution when discontinuities are present [22]. The effect arises when the coefficients of the equation stencil admit negative values. Methods which have strictly non-negative stencil coefficients for a given problem are described as possessing the Convex Monotone Property (CMP) by [22]. These authors also note that that damping is effective in recovering the CMP in Chebyshev polynomial based methods. However, they assert that to maintain the CMP with increasing stage number L, damping must be increased, with a resultant reduction in efficiency such that the extent of the stability domain on the real axis goes as $L^{3/2}$.

In this section, we consider the CMP in the more general case of advection-diffusion problems of the form

$$w_t + aw_x = dw_{xx}, \tag{22}$$

in the presence of an initial discontinuity

$$w(x,\,0) = \begin{cases} w_L & \text{if } x < 0, \\ w_R & \text{otherwise} \end{cases}. \tag{23}$$

The exact solution to eqs. (22) and (23) is given by

$$w(x,\,t) = \frac{1}{2}\left[(w_R + w_L) + (w_R - w_L)\,\mathrm{erf}\left(\frac{x - at}{2\sqrt{dt}}\right)\right]. \tag{24}$$

A uniform mesh is assumed with spacing $h = 0.1$ over the domain $-20 < x < 20$, with Dirichlet boundary conditions imposed using the exact solution. The particular parameters chosen here are $a = 0.2$, $d = 1$, giving a mesh Péclet number of $P = 0.2$. We consider a first order upwind discretization of the advection term and a second order centred discretization of the diffusion term.

The effect of the CMP is evident in fig. 5(a), where solutions from FRKG methods are presented at second order for $M = 11$ over 10 timesteps, with comparisons of the cases $\nu = 0.5$ (corresponding to a Legendre polynomial-based method) and $\nu = 1$. While both cases are stable, imposing $M > 1$ on the initial discontinuity results in severe oscillations in the solutions derived with $\nu = 0.5$, due to an absence of the CMP. We note that the observed oscillatory behaviour is no longer present in the case with $\nu = 1$. Evidence as to the origin of this behaviour may be found by inspecting the sample coefficients from the equation stencil shown over the stable range in fig. 5(b): at several points, the coefficients corresponding to $\nu = 0.5$ become negative, whereas the coefficients for $\nu = 1$ are strictly non-negative over the stable range.

The minimum equation stencil coefficient values are illustrated in fig. 5(c) for different mesh Péclet numbers, $P = 0,\, 0.2$, and stage numbers $M = 7,\, 11$. The suggested characteristics of the CMP for FRKG methods are that it is not maintained for odd $N > 1$, however, for even N, or $N = 1$, the CMP is recovered for some critical value ν_{CMP}. Furthermore, this value of ν_{CMP} is independent of M for $P = 0$, occurring at 0.5 for $N = 1,\, 2$, and subsequently rising slightly for even N. This is consistent with the observed behaviour of the first and second order Legendre polynomial based methods RKL1 and RKL2 [22]. In the case of finite advection with $P = 0.2$, ν_{CMP} appears to rise with M more rapidly than the stable estimate provided by algorithm 1. As a consequence, while increasing ν will significantly dampen the observable staircase artefacts, it is not feasible in practice to fully enforce the CMP for advection-diffusion problems at large M above second order.

We emphasize that the discussions presented here are based on fixed stage number integrations, and that with suitable stepsize control procedures, which will restrict the stage number while discontinuities are diffused, absence of CMP does not generate staircasing in the numerical solution [24].

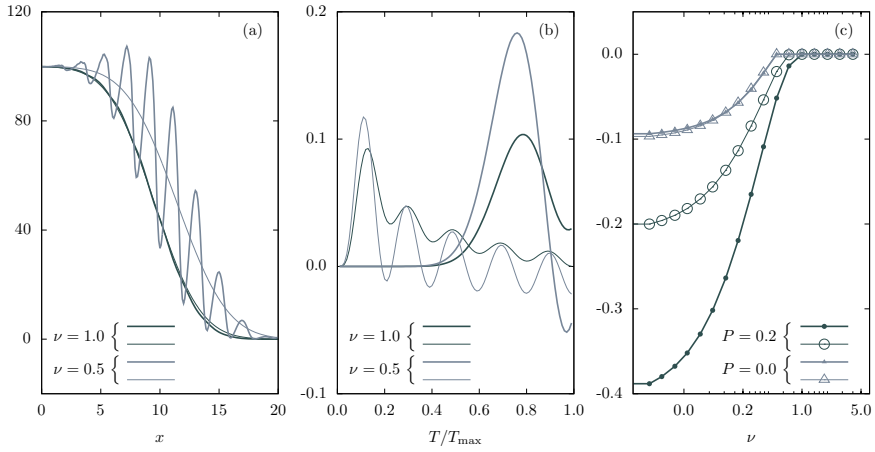

Figure 5: Illustrations of CMP for advection-diffusion problems. Panel (a): Solutions to eqs. (22) and (23) for $\nu = 0.5$ and 1.0 at second order accuracy over 10 timesteps for $M = 11$ at times 4.9710 and 4.1425 respectively, shown in heavy lines. The exact solutions are also shown in fine lines. Panel (b): Examples of the equation stencil coefficients with timesteps up to the stable limit. For the given case, there are 45 coefficients. Heavy lines show the 39th coefficient (the largest negative value). Fine lines show the 29th coefficient. Panel (c): Mimima of equation stencil coefficients as a function of ν for $P = 0.0$ and $P = 0.2$. Heavy lines correspond to $M = 11$ and fine lines to $M = 7$.

3 Brusselator with advection

We consider the Brusselator diffusion-reaction problem [17, 12] extended to include advection,

$$\begin{aligned}
\frac{\partial v}{\partial t} &= \epsilon\left(\frac{\partial^2 v}{\partial {x_1}^2}+\frac{\partial^2 v}{\partial {x_2}^2}\right)+A-(B+1)v+wv^2+\mu\left(U_1\frac{\partial v}{\partial x_1}+U_2\frac{\partial v}{\partial x_2}\right),\\
\frac{\partial w}{\partial t} &= \epsilon\left(\frac{\partial^2 w}{\partial {x_1}^2}+\frac{\partial^2 w}{\partial {x_2}^2}\right)+Bv-v^2w+\mu\left(V_1\frac{\partial w}{\partial x_1}+V_2\frac{\partial w}{\partial x_2}\right),
\end{aligned} \tag{25}$$

with initial conditions $v(0,\,x) = 22x_2(1-x_2)^{1.5}$, $w(0,\,x) = 27x_1(1-x_1)^{1.5}$. Tests are configured with diffusion and reaction parameters $\epsilon = 0.01$, $A = 1.3$, $B = 1$; advection vectors $U = (-0.5, 1)^T$, $V = (0.4, 0.7)^T$; and $\mu = 0.1,\, 0.5,\, 1.0$, corresponding to weak, mild, and moderate advection respectively. Numerical solutions are obtained at $t = 1$ on a uniform mesh with periodic boundary conditions over $0 \le x_1 \le 1$, $0 \le x_2 \le 1$, with $h_1 = h_2 = 1/800$. We employ second order upwind discretizations of the advection terms (corresponding to $\kappa = -1$ under the κ-scheme formalism) with second order centred discretization of the diffusion terms. For the stated advection parameters, the largest mesh Péclet numbers (given by P_2 for species v) are 0.0125, 0.0625, 0.125. The order of the system of ODEs is $2/h^2 = 640000$, with the spectral radius of the Jacobian matrix estimated by $\psi_1^{-1} \approx 5.17 \times 10^4$, 5.36×10^4, 5.60×10^4 respectively. Hence, the problem is moderately stiff and not readily tractable via standard RK techniques.

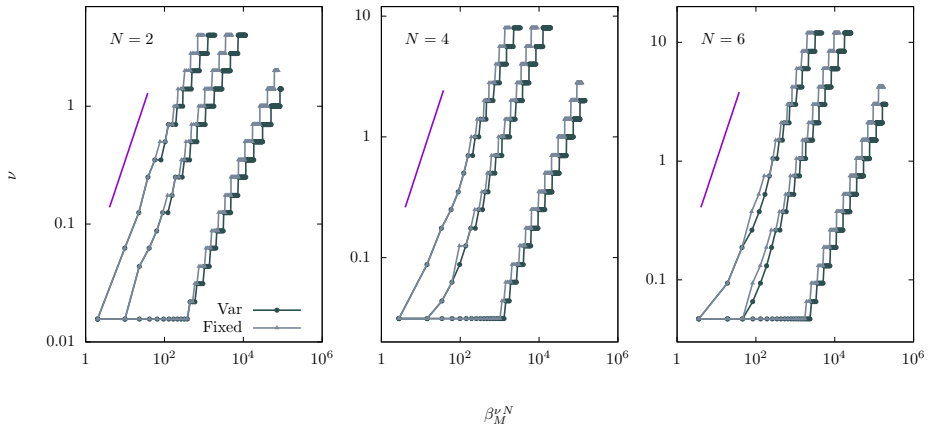

Figure 6: The Gegenbauer parameter, ν, as selected by algorithm 1, plotted against $\beta_M^{\nu N}$ for fixed timestep methods (grey lines) and variable timestep methods using a stepsize controller (black lines). ν ranges from $N/128$ to $2N$ at 17 logarithmically uniformly spaced values. The methods corresponding to the weak, mild and moderate advection ($P_2 = 0.0125, 0.0625, 0.125$) are shown from left to right. Guidelines are linear in $\beta_M^{\nu N}$.

Complex splitting of the form given by eq. (18) is employed at fourth and sixth orders using coefficients from [6] and [4] (also presented in table 1). Reaction terms are treated using standard adaptive RK integrators from the GNU Scientific Library [8].

The methods implemented in the tests are selected via algorithm 1, for both fixed stepsizes, and stepsizes controlled via local error estimates. The associated Gegenbauer

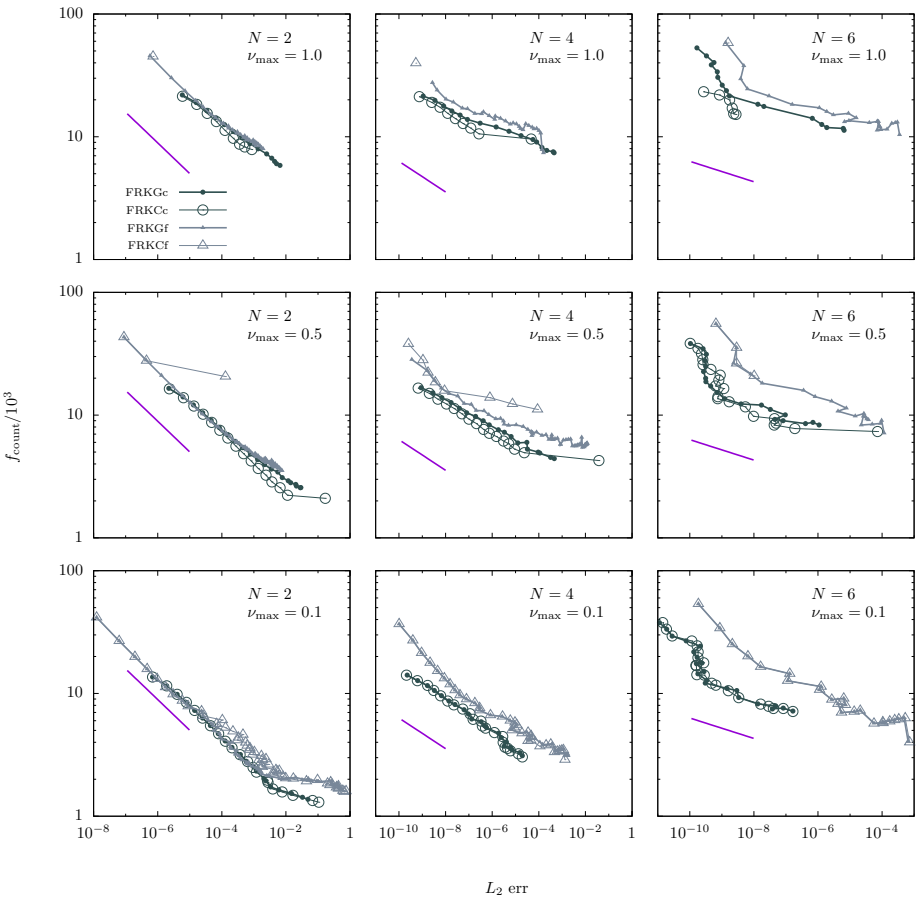

Figure 7: Work-precision diagram for Brusselator problem with advection. Second order unsplit method results are shown in the left column, with fourth and sixth order split method results in the second and third columns. Mesh Péclet numbers, from bottom row to top, are 0.0125, 0.0625, 0.125, corresponding to weak, mild, and moderate advection respectively. Data from FRKG method integrations with stepsize control (FRKGc) and fixed stepsizes (FRKGf) are shown as filled circles and filled triangles respectively. Unfilled circles and triangles correspond to results from Chebyshev polynomial based methods with stepsize control (FRKCc) and fixed stepsizes (FRKCf). Guidelines shown are proportional to $(\text{err})^{-1/2N}$.

parameter, ν, is shown as a function of the maximum timestep size in fig. 6. It can be seen that increasing ν up to the maximum value of $2N$ extends the stable stepsize by approximately two orders of magnitude, whilst continuing to contain the domain of the methods' eigenvalues within the ellipse approximations given by eq. (13). The realized timesteps represent a gain in efficiency over standard RK integrations of $1.5M$, down to $0.5M$ with increasing ν. Typical RKG stability domains for the integration of the linear parts of section 3 encountered at fourth order may be seen in fig. 2.

Work-precision test results for the Brusselator problem with advection are shown in fig. 7 for unsplit second order and split higher order FRKG methods. Cases are considered with both fixed stepsize over the full integration, and variable stepsizes controlled by the error estimation procedure described in section 2.5. (Indicative numerical data for the tests are also provided in appendix B.) For comparison, results from integrations carried out with Chebyshev polynomial based FRKC methods are also presented.

Stepsize control is clearly an effective counter-measure to the growth of numerical instabilities. In fact, where the advective term is weak-to-mild, or where the internal stage number is low, the stepsize controller is adequate in managing instability growth arising due to eigenvalues with imaginary parts extending into the exterior of the method's stability domain. For large internal stage number with large advection, as seen in the top row of fig. 7, instability develops excessively within a single timestep and stepsize control is no longer effective. We note that splitting appears to generate a complex error behaviour which is significantly overestimated by eq. (19). This is managed by adopting tolerance parameters *atol* and *rtol* for the controller at values larger than the target precision in eq. (19). We find that integrations utilizing stepsize control are more efficient in general than fixed stepsize integrations, particularly for higher order split methods.

In the absence of stepsize control, the Chebyshev polynomial based FRKC methods with weak damping are largely ineffective, with the exception of cases where the advection term is very small (see the bottom row of fig. 7). As noted by [27], increasing damping improves the performance of these methods. In contrast, RKG methods rely solely on the natural characteristics of the Gegenbauer polynomials forming the underlying RKG stability polynomials and do not require explicit damping procedures.

4 Conclusions

In this paper, we have presented the class of Runge–Kutta–Gegenbauer (RKG) stability polynomials in closed form to arbitrarily high order of accuracy. The RKG polynomials of order N, and degree $L = MN$, comprise a linear combination of Gegenbauer polynomials of degree kM, for $k = 1, \ldots, N$, and common Gegenbauer parameter. The particular weighting of the combination is chosen to conform to the linear order conditions, subject to maximizing the extent of the stability domain along the negative real axis, which scales as L^2. Crucially, for the consideration of systems of mixed hyperbolic-parabolic type, the domain extends in the imaginary direction as an increasing function of the Gegenbauer parameter.

We have demonstrated the construction of Factorized Runge–Kutta–Gegenbauer (FRKG) explicit methods to high order, consisting of ordered sequences of forward Euler stages with complex-valued stepsizes. The algorithm implemented in ordering the L stages of a given method prevents internal amplification factors from overwhelming available numerical precision by bounding their magnitudes to $10L^2$.

RKG stability polynomials are shown to be effective in the construction of high order explicit methods for mildly stiff advection-diffusion problems with moderate ($\lesssim 1$) mesh Péclet numbers. [6]

[6] An implementation of second order FRKG methods is available as Electronic Supplemen-

Acknowledgments

The author wishes to acknowledge the DJEI/DES/SFI/HEA Irish Centre for High-End Computing (ICHEC) for the provision of computational facilities and support.

Data Statement

A Splitting methods

Table 1: Complex operator splitting parameters, corresponding to eq. (18), at orders $N = 4, 6$ [6, 4, 1]. For each quoted value of N, the rows show: j, the index for a distinct sweep, with $\mathrm{Re}(T_j)$ and $\mathrm{Im}(T_j)$, the real and imaginary components of the associated timescale; J (in the last row), the number of distinct sweeps required; and the sequence of labels j identifying $k_1 \cdots k_J$.

N	j	$\mathrm{Re}(T_j)$ $\mathrm{Im}(T_j)$
	J	$k_1 \cdots k_J$
4	1	1/4 *0*
	2	1/10 *−1/30*
	3	4/15 *2/15*
	4	4/15 *−1/5*
	9	2 1 3 1 4 1 3 1 2
6	1	0.0625 *0.0*
	2	0.024 694 876 087 018 064 640 910 864 996 842 247 838 60 *−0.007 874 795 562 906 877 058 171 577 949 526 942 163 20*
	3	0.063 813 474 021 302 699 779 366 304 188 200 146 963 20 *0.035 365 761 034 143 327 804 629 404 649 714 741 812 70*
	4	0.068 425 094 030 316 441 970 397 007 821 744 684 058 50 *−0.062 262 244 450 748 676 995 332 540 644 447 596 046 10*
	5	0.088 047 701 092 267 837 626 997 195 869 408 667 577 20 *0.045 473 871 502 298 704 383 762 549 187 977 426 444 69*
	6	0.023 689 611 129 847 060 696 141 912 470 009 364 325 33 *0.009 624 326 064 089 624 057 698 035 290 637 306 663 95*
	7	0.042 729 722 386 773 382 202 964 300 577 074 218 553 88 *−0.033 994 403 923 957 610 554 083 948 457 844 358 264 99*
	8	0.122 334 686 316 845 772 960 428 517 001 962 563 078 80 *−0.010 435 859 079 752 510 669 380 827 100 590 549 551 78*
	9	0.041 898 432 829 693 886 043 536 850 607 262 239 764 26 *0.069 362 492 631 696 384 275 158 174 307 144 262 130 30*
	10	0.048 732 804 211 869 708 158 514 092 934 991 735 680 80 *−0.090 518 296 429 724 730 488 558 538 566 128 582 051 30*
	33	2 1 3 1 4 1 5 1 6 1 7 1 8 1 9 1 10 1 9 1 8 1 7 1 6 1 5 1 4 1 3 1 2

tary Material.

B Work-precision data tables for Brusselator with advection.

Table 2: Second order ($N = 2$) FRKG method results for the Brusselator problem with advection detailed in section 3. Columns show: L_2 (and L_∞) errors; count of function evaluations, f_{count}; number of timesteps attempted, Timesteps; and the maximum value of M required, $M_{\max}$. Weak, mild, and moderate advection data ($\mu = 0.1$, 0.5, 1.0) are presented, from bottom to top respectively. For rows where values of *atol* are given, the stepsize controller is implemented and f_{count} values refer to accepted (rejected) function evaluations.

atol	L_2 (L_∞) error	f_{count} (rej)	Timesteps	$M_{\max}$
		$N = 2$		
		$\mu = 1.0$		
10^{-2}	5.09e-03 (3.06e-02)	6030 (0)	52	51
10^{-3}	1.35e-03 (8.61e-03)	7975 (0)	154	51
10^{-4}	1.22e-04 (7.71e-04)	12559 (0)	533	20
10^{-5}	5.86e-06 (3.70e-05)	21898 (0)	1766	9
	1.12e-04 (4.84e-04)	12870	322	10
	1.08e-03 (4.61e-03)	9120	76	30
		$\mu = 0.5$		
1	2.81e-02 (2.06e-01)	2561 (0)	14	126
10^{-1}	1.40e-02 (9.53e-02)	2814 (0)	20	126
10^{-2}	3.49e-03 (2.28e-02)	3628 (0)	44	122
10^{-3}	4.52e-04 (2.92e-03)	5423 (0)	124	48
10^{-4}	4.66e-05 (3.00e-04)	9065 (0)	382	21
10^{-5}	2.22e-06 (1.43e-05)	16525 (0)	1209	11
	3.75e-05 (1.74e-04)	9230	231	10
	7.99e-04 (3.67e-03)	5010	42	30
	1.51e-03 (6.85e-03)	4480	28	40
	2.11e-03 (9.37e-03)	4400	22	50
	4.29e-03 (1.89e-02)	3780	16	60
	3.94e-03 (1.74e-02)	4130	15	70
	6.59e-03 (2.90e-02)	3520	11	80
		$\mu = 0.1$		
1	4.87e-02 (4.00e-01)	1373 (0)	14	105
10^{-1}	3.32e-03 (1.55e-02)	1733 (0)	21	80
10^{-2}	9.91e-04 (5.46e-03)	2517 (0)	42	65
10^{-3}	1.26e-04 (6.48e-04)	4106 (0)	105	38
10^{-4}	1.42e-05 (7.12e-05)	7277 (0)	307	23
10^{-5}	6.90e-07 (3.44e-06)	13609 (0)	950	13
	8.91e-06 (4.74e-05)	7870	197	10
	2.02e-04 (8.44e-04)	4000	50	20
	6.78e-04 (3.04e-03)	2760	23	30
	1.78e-03 (8.67e-03)	2200	14	40
	2.75e-01 (5.07e+00)	1950	10	50

Table 3: Fourth and sixth order ($N = 4$, 6) split FRKG method sample results.

atol	L_2 (L_∞) error	f_{count} (rej)	Timesteps	$M_{\max}$
		$N = 4$		
		$\mu = 1.0$		
10^{-2}	1.37e-04 (6.68e-04)	8178 (0)	22	51
10^{-3}	5.24e-06 (3.16e-05)	10672 (410)	49	49
10^{-4}	4.18e-08 (2.74e-07)	14886 (130)	107	46
10^{-5}	1.09e-09 (8.14e-09)	21444 (0)	235	24
	8.74e-08 (3.48e-07)	16900	65	10
		$\mu = 0.5$		
10^{-1}	1.10e-04 (3.37e-04)	4912 (0)	16	123
10^{-2}	1.26e-05 (4.49e-05)	5936 (0)	24	113
10^{-3}	7.47e-07 (2.75e-06)	7996 (326)	45	73
10^{-4}	3.92e-08 (1.44e-07)	11244 (116)	85	43
10^{-5}	8.93e-10 (3.27e-09)	16718 (0)	177	27
	6.21e-08 (2.52e-07)	12420	48	10
	1.07e-05 (4.55e-05)	8320	15	20
	7.99e-05 (3.44e-04)	7260	9	30
	4.93e-04 (1.83e-03)	6400	6	40
	2.77e-03 (1.11e-02)	5600	4	50
	6.29e-03 (2.22e-02)	5760	4	60
		$\mu = 0.1$		
1	1.95e-05 (6.33e-05)	3108 (0)	12	101
10^{-1}	3.22e-06 (1.04e-05)	3714 (0)	16	77
10^{-2}	5.69e-07 (2.00e-06)	4842 (426)	26	66
10^{-3}	9.29e-08 (3.46e-07)	6500 (310)	41	53
10^{-4}	6.59e-09 (2.47e-08)	9486 (114)	74	40
10^{-5}	2.26e-10 (8.40e-10)	14024 (82)	149	28
	2.55e-08 (1.08e-07)	10640	41	10
	1.12e-05 (4.07e-05)	5600	11	20
	1.45e-04 (6.48e-04)	3840	5	30
	1.60e-03 (5.10e-03)	3200	3	40
		$N = 6$		
		$\mu = 1.0$		
10^{-2}	1.31e-06 (9.22e-06)	12584 (0)	18	49
10^{-3}	1.72e-09 (6.92e-09)	20164 (1334)	47	49
10^{-4}	7.18e-10 (3.20e-09)	33140 (644)	101	48
10^{-5}	1.67e-10 (7.54e-10)	53002 (0)	221	25
	3.03e-06 (1.62e-05)	15080	9	10
	3.52e-04 (1.23e-03)	10380	2	30
		$\mu = 0.5$		
1	1.08e-06 (5.30e-06)	8308 (0)	12	123
10^{-1}	4.47e-08 (1.66e-07)	9224 (0)	15	123
10^{-2}	3.98e-09 (1.49e-08)	12328 (0)	23	123
10^{-3}	3.17e-10 (1.06e-09)	17534 (1188)	44	79
10^{-4}	2.95e-10 (1.25e-09)	26278 (1484)	85	44
	7.68e-06 (3.09e-05)	11400	7	10
	9.64e-05 (3.53e-04)	9040	3	20
		$\mu = 0.1$		
1	1.65e-07 (1.00e-06)	7188 (0)	12	101
10^{-1}	1.32e-08 (8.79e-08)	8222 (0)	15	91
10^{-2}	3.03e-10 (1.27e-09)	11090 (1062)	25	67
10^{-3}	1.60e-10 (5.95e-10)	15694 (1798)	42	59
10^{-4}	2.14e-10 (8.17e-10)	22804 (1468)	74	40
10^{-5}	1.13e-11 (3.81e-11)	36440 (1250)	146	29
	1.12e-06 (3.72e-06)	10720	7	10
	1.05e-04 (4.79e-04)	5880	2	20

C Supplementary material

Supplementary material related to this article can be found online at `https://doi.org/10.1016/j.jcp.2019.03.001`.

References

[1] Complex coefficients of several splitting and composition methods for integrating in time several linear and nonlinear evolution equations. http://www.gicas.uji.es/Research/splitting-complex.html.

[2] Abdulle, A. Fourth order Chebyshev methods with recurrence relation. *SIAM Journal on Scientific Computing 23*, 6 (2002), 2041–2054.

[3] Bini, D. A., and Fiorentino, G. Design, analysis, and implementation of a multiprecision polynomial rootfinder. *Numerical Algorithms 23*, 2 (2000), 127–173.

[4] Blanes, S., Casas, F., Chartier, P., and Murua, A. Optimized high-order splitting methods for some classes of parabolic equations. *Mathematics of Computation 82*, 283 (2013), 1559–1576.

[5] Butcher, J. C. *Numerical methods for ordinary differential equations. 2nd revised ed.*, 2nd revised ed. ed. Hoboken, NJ: John Wiley & Sons, 2008.

[6] Castella, F., Chartier, P., Descombes, S., and Vilmart, G. Splitting methods with complex times for parabolic equations. *BIT Numerical Mathematics 49* (2009), 487–508.

[7] Fousse, L., Hanrot, G., Lefèvre, V., Pélissier, P., and Zimmermann, P. MPFR: A multiple-precision binary floating-point library with correct rounding. *ACM Transactions on Mathematical Software (TOMS) 33*, 2 (2007), 13.

[8] Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Booth, M., and Rossi, F. *GNU Scientific Library Reference Manual*, 3rd ed. Network Theory Ltd., 2009.

[9] Granlund, T., and GMP development team. GNU Multiple Precision Arithmetic library, 2016. https://gmplib.org.

[10] Gustafsson, K. Control-theoretic techniques for stepsize selection in implicit runge-kutta methods. *ACM Transactions on Mathematical Software (TOMS) 20*, 4 (1994), 496–517.

[11] Hairer, E., Nørsett, S., and Wanner, G. Solving ordinary differential equations I: Nonstiff problems. *Springer series in computational mathematics 8* (1993).

[12] Hairer, E., and Wanner, G. Solving ordinary differential equations II: Stiff and differential-algebraic problems. *Springer series in computational mathematics 14* (1996).

[13] Hansen, E., and Ostermann, A. High order splitting methods for analytic semigroups exist. *BIT Numerical Mathematics 49* (2009), 527–542.

[14] Hundsdorfer, W., and Verwer, J. G. *Numerical solution of time-dependent advection-diffusion-reaction equations*, vol. 33. Springer, 2003.

[15] Lebedev, V. How to solve stiff systems of differential equations by explicit methods. *Numerical methods and applications* (1994), 45–80.

[16] Lebedev, V. I. Explicit difference schemes for solving stiff problems with a complex or separable spectrum. *Computational mathematics and mathematical physics 40*, 12 (2000), 1729–1740.

[17] Lefever, R., and Nicolis, G. Chemical instabilities and sustained oscillations. *Journal of theoretical Biology 30*, 2 (1971), 267–284.

[18] Lubich, C., and Ostermann, A. Interior estimates for time discretizations of parabolic equations. *Applied numerical mathematics 18*, 1 (1995), 241–251.

[19] Martn-Vaquero, J., and Kleefeld, B. Extrapolated stabilized explicit rungekutta methods. *Journal of Computational Physics 326* (2016), 141155.

[20] Medovikov, A. A. High order explicit methods for parabolic equations. *BIT Numerical Mathematics 38*, 2 (1998), 372–390.

[21] Meyer, C. D., Balsara, D. S., and Aslam, T. D. A second-order accurate Super TimeStepping formulation for anisotropic thermal conduction. *Monthly Notices of the Royal Astronomical Society 422*, 3 (2012), 2102–2115.

[22] Meyer, C. D., Balsara, D. S., and Aslam, T. D. A stabilized Runge-Kutta-Legendre method for explicit super-time-stepping of parabolic and mixed equations. *Journal of Computational Physics 257* (2014), 594–626.

[23] O'Sullivan, S. A class of high-order Runge-Kutta-Chebyshev stability polynomials. *Journal of Computational Physics 300* (2015), 665 – 678.

[24] O'Sullivan, S. Factorized Runge-Kutta-Chebyshev methods. *Journal of Physics: Conference Series 837*, 1 (2017), 012020.

[25] Sommeijer, B., Shampine, L., and Verwer, J. RKC: an explicit solver for parabolic PDEs. *Journal of Computational and Applied Mathematics 88*, 2 (1997), 315–326.

[26] Van Leer, B. Upwind-difference methods for aerodynamic problems governed by the euler equations. *Lectures in applied mathematics 22*, Part 2 (1985), 327–336.

[27] Verwer, J. G., Sommeijer, B. P., and Hundsdorfer, W. RKC time-stepping for advection–diffusion–reaction problems. *Journal of Computational Physics 201*, 1 (2004), 61–79.

[28] Watts, H. A. Step size control in ordinary differential equation solvers. *Transactions of the Society for Computer Simulation 1* (1984), 15–25.

[29] Wesseling, P. *Principles of computational fluid dynamics*, vol. 29. Springer Science & Business Media, 2009.