IOP Conf. Series: Journal of Physics: Conf. Series **837** (2017) 012020 doi:10.1088/1742-6596/837/1/012020

Factorized Runge–Kutta–Chebyshev Methods

Stephen O'Sullivan

School of Mathematical Sciences, Dublin Institute of Technology, Kevin Street, Dublin 8, Ireland

E-mail: `stephen.osullivan@dit.ie`

Abstract. The second-order extended stability Factorized Runge–Kutta–Chebyshev (**FRKC2**) explicit schemes for the integration of large systems of PDEs with diffusive terms are presented. The schemes are simple to implement through ordered sequences of forward Euler steps with complex stepsizes, and easily parallelised for large scale problems on distributed architectures. Preserving 7 digits for accuracy at 16 digit precision, the schemes are theoretically capable of maintaining internal stability for acceleration factors in excess of 6000 with respect to standard explicit Runge-Kutta methods. The extent of the stability domain is approximately the same as that of **RKC** schemes, and a third longer than in the case of **RKL2** schemes. Extension of **FRKC** methods to fourth-order, by both complex splitting and Butcher composition techniques, is also discussed.
A publicly available implementation of **FRKC2** schemes may be obtained from `maths.dit.ie/frkc`

1. Introduction

Factorized Runge–Kutta–Chebyshev (**FRKC**) methods are well suited to the numerical integration of problems where diffusion limits the efficiency of standard explicit techniques. In general, such systems of PDEs may be presented as semi-discrete ordinary differential equations of the form

$$w' = f(t, w). \tag{1}$$

Here, the associated Jacobian is assumed to have negative eigenvalues lying close to the real axis, a good approximation for many systems of interest in astrophysical contexts.

The main use of extended stability Runge–Kutta (ESRK) methods is to fill the gap between unconditionally stable but operationally complex implicit methods, and simply implemented explicit schemes which suffer from stability constraints for stiff problems. ESRK methods are particularly useful for problems involving diffusion, where the work required by standard explicit techniques goes as the inverse square of the mesh spacing, while for extended stability methods it goes as the inverse mesh spacing. ESRK explicit schemes may be broadly divided into factorized and recursive types.

Factorized ESRK methods are particularly straightforward to implement at second-order, consisting solely of forward Euler steps. At orders above two, splitting methods or, alternatively, additional finishing stages are required for nonlinear problems. First considered by [1, 2, 3], factorized ESRK methods fell out of common usage for some time, until revived in 1996 as SuperTimeStepping [4] at first-order, and later extended to second-order applications in

astrophysical simulations by means of Richardson extrapolation [5, 6]. **DUMKA** methods exist at third- and fourth-order [7].

The perennial problem with the factorized formalism has been that, when a very large number of stages is used, the internal amplification factors can easily drown numerical precision. Factorized methods have, as a result, largely taken a back seat to recursive methods which manage internal stability by mapping the three-term recurrence relations for orthogonal polynomials to the stability polynomials [8]. Recursive methods have been developed up to fourth-order [9, 10, 11, 12, 13, 14, 15].

In the following, a formulation of factorized methods is presented which has high internal stability, and is more straightforward to implement than recursive methods, and demonstrates comparable efficiency.

2. FRKC stability polynomials

Stability polynomials form the backbone of ESRK numerical schemes and encapsulate the linear stability properties. Linearising a system of semi-discrete ODEs

$$w' = Aw, \tag{2}$$

the FRKC stability polynomial [16] is obtained by seeking a polynomial of degree L which yields a forward Euler scheme of order N for linear problems through its roots via

$$W^L = W^0 + T\sum_{l=1}^{L} a_l f(W^{l-1}) \quad W^0 = w^n,\, w^{n+1} = W^L. \tag{3}$$

The M-th stability polynomial of order N, with $L = MN$, determined via $w^{n+1} = R_M^N(TA)w^n$, must match the first $N+1$ terms in the Taylor expansion of the evolution operator

$$R_M^N(TA) = \mathrm{e}^{TA} + \mathcal{O}(TA)^{N+1}. \tag{4}$$

Equivalently, the linear order conditions may be expressed as constraints on the derivatives of the stability polynomial evaluated at zero:

$$R_M^{N\,(n)}(0) = 1, \quad n = 1, \cdots, N. \tag{5}$$

In addition, stability requires that the polynomial is bounded according to

$$|R_M^N(z)| \leq 1 \quad \forall \quad z = T\lambda. \tag{6}$$

The objective is to determine a closed form for the polynomial such that the extent of the stability domain along the negative real axis β is as great as possible.

It is shown in [16] that the **FRKC** stability polynomial of rank N, and degree L, is a sum of Chebyshev polynomials of the first kind given by

$$B_M^N(z) = d_0^N + 2\sum_{k=1}^{N} d_k^N C_{kM}(z), \tag{7}$$

where C_{kM} is the Chebyshev polynomial of the first kind of degree kM, and the coefficients d_k are determined through the linear order conditions given by Equation 5. The resultant scheme follows immediately from the roots of the polynomial, ζ_l, with coefficients given by

$$a_l = \frac{1}{M^2\alpha_M}\,\frac{1}{1-\zeta_l}. \tag{8}$$

The dependency of β on L is presented in Figure 1 at second-order ($N = 2$). While the optimal value for $\beta/L^2\beta_{RK2}$ is 0.41 [17], where β_{RK2} is the conventional second order Runge–Kutta stability limit, values of 0.330, 0.333, and 0.25 are obtained for **FRKC2**, **RKC** [8], and **RKL2** [15] respectively.

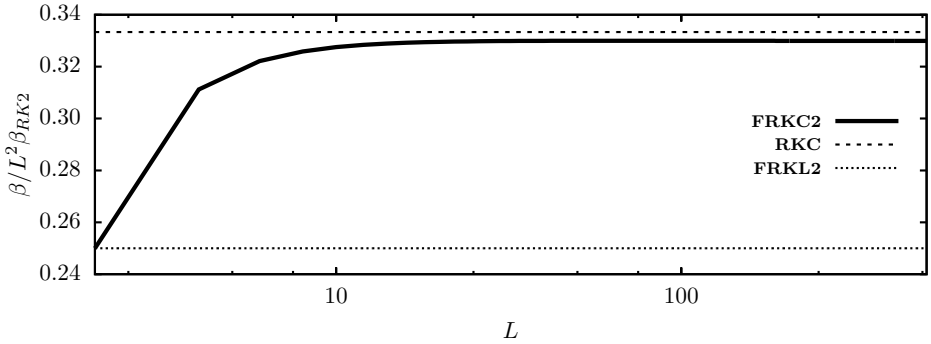


Figure 1: Extent of real stability domain at second-order for **FRKC2** (solid line). For large values of stage-number L, the scheme has a stability domain which is $0.33L^2$ times the extent of the reference second-order Runge–Kutta scheme. Also shown are the corresponding values for **RKC** (dashed line) and **RKL2** (dotted line).

2.1. Damping

Along the real axis on the interior of the stability domain of the stability polynomial, there are points which are marginally stable, as shown in Figure 2. This is remedied by introducing the damping parameter $\nu \equiv \nu_0/N$ via

$$a_l = \frac{1}{(1-\nu)M^2\alpha_M}\,\frac{1-\mu_l}{1-(1-2\mu_l)\zeta_l}, \tag{9}$$

and again enforcing the order conditions given by Equation 5 via Newton-Raphson iteration over the parameters μ_l, which consist of N distinct values, each repeated M times. As a result, the marginally stable maxima in $|R|$ along the real axis are scaled by $\sim (1-\nu_0)$ at the expense of reducing the extent of the stability domain β along the real axis by approximately $(1-\nu)$.

The damping process may also be used to make the scheme applicable to problems with small hyperbolic components, with Péclet numbers $Pe \lesssim 10$. For the case $Pe \approx 10$, illustrated in Figure 2, there is a 25% loss in the extent of the stability domain observed.

2.2. Internal stability

Internal instability may be caused when the product of any of the possible sub-sequences of steps act to generate large values which drown out numerical precision. Following the idea of Lebedev [18, 19], but with a more effective approach, the timesteps are ordered to *approximately* minimise $\mathcal{Q}$, where

$$\mathcal{Q} \equiv \max(\mathcal{Q}_{j,\,k}(x)), \quad 1 \le j \le k \le L, \quad x \in [-\beta,\, 0], \tag{10}$$

is the maximum over the internal amplification factors defined by

$$\mathcal{Q}_{j,\,k}(x) = \prod_{l=j}^{k} |1 + a_l x|. \tag{11}$$

While the optimal value of $\mathcal{Q}$ is approximately L^2, for the purposes of constructing a stabilization algorithm, a practical upper bound of $10L^2$ is chosen. In order to maintain this bound, the estimated maximum amplification factor $\overline{\mathcal{Q}}$ is held to a minimum while l runs from 1 to L, where

$$\overline{\mathcal{Q}} \equiv \left\| \max\left(\prod_{j=1}^{l} v_{j,\,k},\ \prod_{j=l+1}^{L/2} v_{j,\,k} \right) \right\|_1 . \tag{12}$$

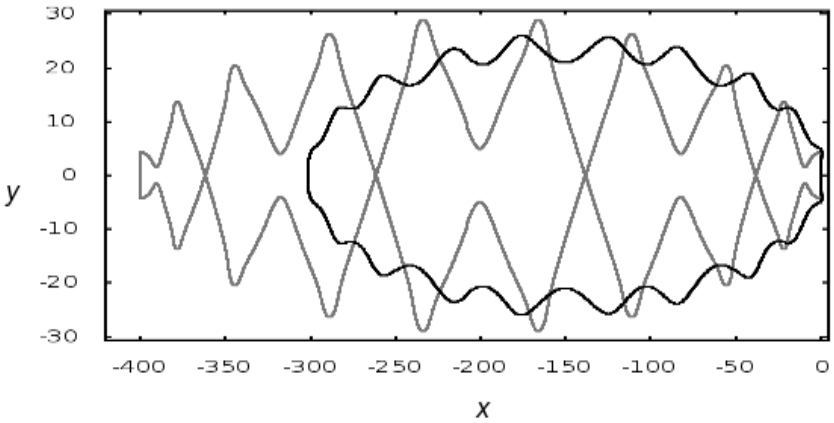


Figure 2: Stability domain $|R(x+Iy)| = 1$ for **FRKC** polynomial without damping (grey line) and with sufficient damping applied for mixed hyperbolic-parabolic problems with $Pe \approx 10$ (black line). The extent of the stability domain along the real axis is contracted by approximately 25% in the latter case.

The amplification factors $v_{j,k}$ are defined by $v_{j,k} = |1 + a_j x_k|$, where $x_k \in [\overline{\beta}, 0]$ are L uniformly distributed values over the reduced range $\overline{\beta} = (1 - nC)\beta$, with $C = 10^{-4}$. Initially, $n = 1$, however, in a limited number of cases, the process is repeated with n incremented until the required bound is satisfied. In this work, the mean value of n for the second-order schemes was found to be 1.5. Figure 3 shows the realised values of $\mathcal{Q}$ obtained via the stabilization algorithm for second-order schemes. Preserving 7 digits for accuracy, a scheme consisting of 10^4 stages is therefore theoretically viable in a numerical integration carried out to 16 digit precision.

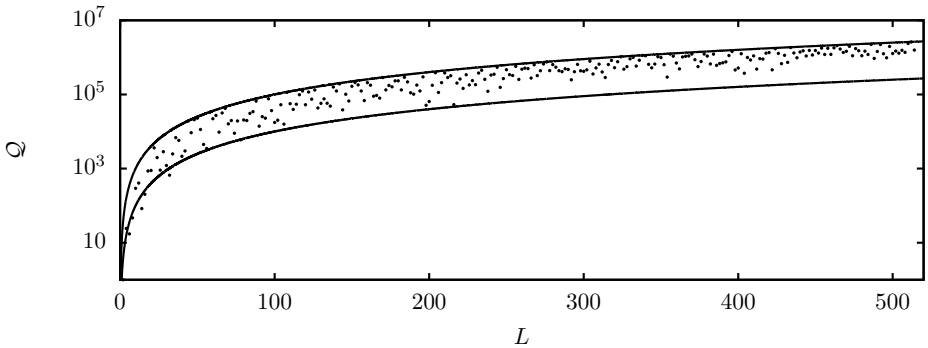
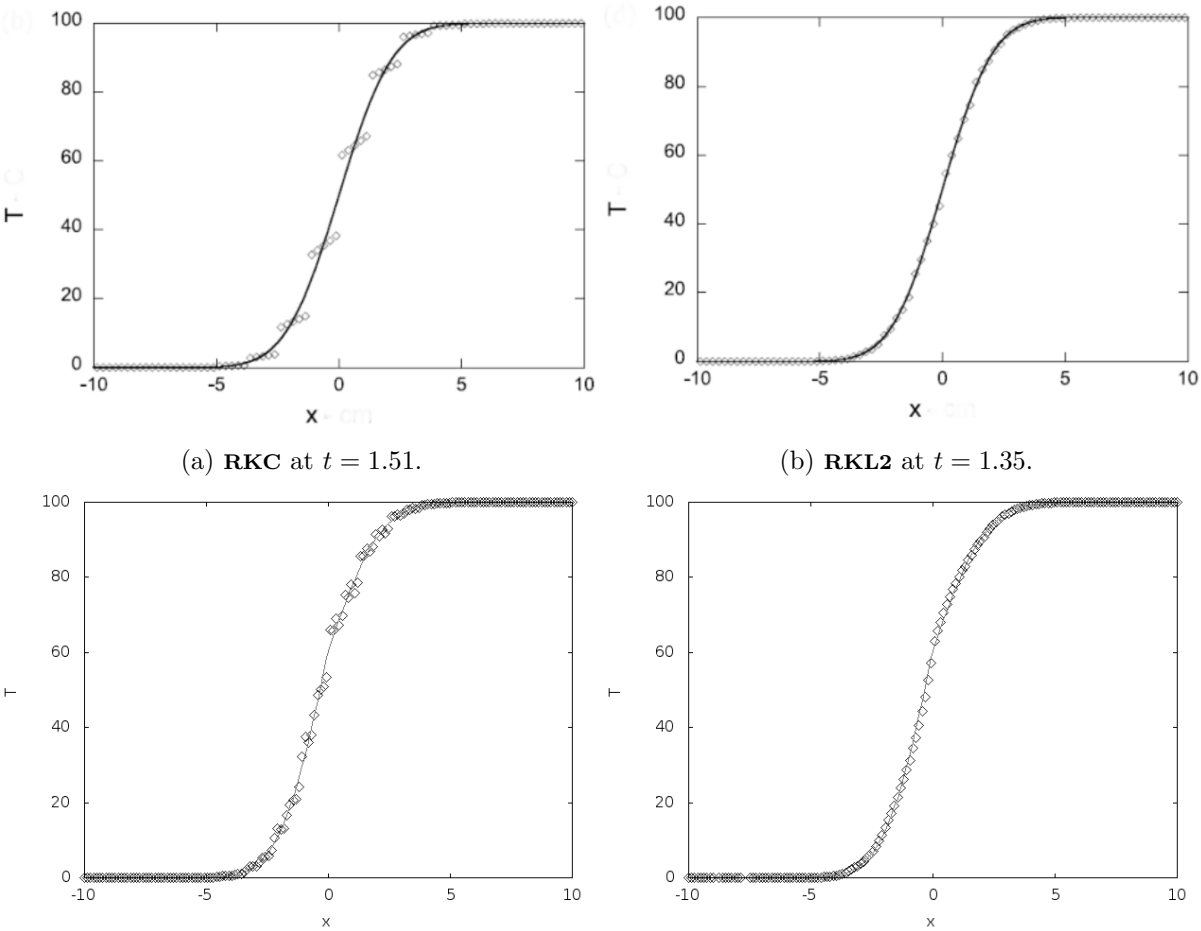


Figure 3: The maximum realised internal amplification factor $\mathcal{Q}$ as a function of L for second-order schemes. Guidelines are at L^2 and $10L^2$.

2.3. Convex Monotone Property

The convex monotone property (CMP), relevant to problems with spatially varying diffusion coefficients, was recently considered by [15]. Figure 4 shows solutions obtained for a problem describing two materials at different temperatures and with different diffusion coefficients placed into contact. For Chebyshev polynomial-based schemes such as **RKC** and **FRKC2**, features in the solution associated with failure to meet the CMP are evident if the schemes are forced to take

large steps initially (Figures 4a and 4c). However, with error control implemented as discussed in Section 3, the features do not appear (Figure 4d). The **RKL2** scheme maintains the CMP naturally (Figure 4b).



(a) **RKC** at $t = 1.51$.

(b) **RKL2** at $t = 1.35$.

(c) **FRKC2** at $t = 1.24$ without error control.

(d) **FRKC2** at $t = 1.24$ with error control.

Figure 4: Heat conduction between materials of different thermal conductivity illustrating influence of the convex monotone property (CMP) [15].

3. FRKC2 schemes

A system $w' = f(w)$ is assumed such that the associated Jacobian has an eigenvalue of maximum magnitude $|\lambda|_{\max}$. Then, given a numerical solution w^n at some time index n, $L = 2M$ stages W^l (for $l = 0, \cdots, L$) are evaluated such that $W^0 = w^n$ and a second-order accurate solution $w^{n+1} = W^L$ is obtained a time T later. The intermediate stages of the scheme are determined via the Euler steps

$$W^{l+1} = W^l + Ta_l f(W^l). \tag{13}$$

Error control is straightforward since a first-order solution is available at no additional cost in function evaluations. This first-order solution $\widehat{W}^L$ is obtained by considering only the real

parts of a_l and $f(W^l)$. Setting $\widehat{W}^0 = W^0$,

$$\widehat{W}^{l+1} = \widehat{W}^l + T\,\mathrm{Re}(a_l)\,\mathrm{Re}(f(W^l)). \tag{14}$$

The error, scaled to a specified tolerance TOL, is estimated using

$$\|err\| = \left\| \frac{|W^{l+1} - \widehat{W}^{l+1}|}{TOL(1 + \max(|W^{l+1}|,\, |\widehat{W}^{l+1}|)} \right\|. \tag{15}$$

If $\|err\| > 1$, the step is rejected and retried with T scaled by $SAFE/\sqrt{\|err\|}$. Otherwise, a predictive controller is used to determine the subsequent timestep calibrated to the required tolerance via

$$T^{n+1} = \left(\frac{SAFE}{\sqrt{\|err^n\|}}\right)\left(\frac{T^n}{T^{n-1}}\right)\sqrt{\frac{\|err^{n-1}\|}{\|err^n\|}}, \tag{16}$$

using values of T and $\|err\|$ from previous timesteps. $SAFE$ is a safety factor chosen with a value 0.8 here. The reader is referred to [20] for further details of error control procedures.

*3.1. **FRKC2** public code*

A freely available C implementation of the second-order **FRKC2** schemes may be accessed at `maths.dit.ie/frkc`. The files frkc2core.c and frkc2user.c provide the code for internal calculations required by the **FRKC2** scheme and the code specific to the particular problem respectively. For a given value of M, up to 257, the extent of the stability domain along the real axis, $\overline{\beta}$, and the maximum realised internal amplification factor $\mathcal{Q}$ (see also Figure 3) are given on line $3M - 2$ of frkc2arks.dat. The real and imaginary parts of a_l are recorded on lines $3M - 2$ and $3M$ respectively.

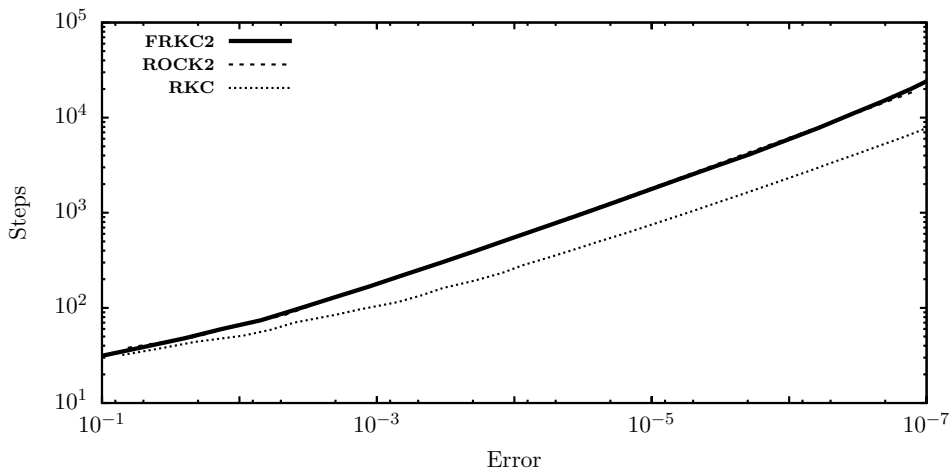


Figure 5: Efficiency comparison for the two-dimensional Brusselator problem for **FRKC2**, **ROCK2**, and **RKC**. (The lines for **FRKC2** and **ROCK2** are almost coincident.)

The default problem provided in frkc2user.c is a two-species reaction diffusion Brusselator

$$\begin{aligned} \partial v/\partial t &= 0.02\left(\partial^2 v/\partial {x_1}^2 + \partial^2 v/\partial {x_2}^2\right) + 1 - 4v + wv^2, \\ \partial w/\partial t &= 0.02\left(\partial^2 w/\partial {x_1}^2 + \partial^2 w/\partial {x_2}^2\right) + 3v - v^2 w, \\ v(0,\, x) &= 1 + \sin(2\pi x) \quad w(0,\, x) = 3 + \cos(2\pi y), \end{aligned} \tag{17}$$

which possesses a spectral radius of approximately 6400 for a 200×200 mesh. The initial state is a perturbation of the equilibrium solution [21]which is given by $v = 1$, $w = 3$. Figure 5 shows the number of steps required to attain a given error in the solution for **FRKC2**, **ROCK2** [22], and **RKC**. It is evident that, for a given precision, there is little difference in the number of steps required by **FRKC2** and **ROCK2**. At higher degrees of acceleration (fewer steps), the difference between the three schemes is negligible.

4. FRKC4 scheme

4.1. Complex splitting

Above second-order, nonlinear order conditions are present which require additional consideration. One approach, given a semi-linear parabolic (reaction-diffusion) equation of the form $w' = Aw + f_B(w)$, is to split the nonlinear part $f_B(w)$, which is typically easily integrated, from the linear diffusion terms Aw. For orders above two, this requires complex timesteps [23, 24] and may be prescribed in the form

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

Figure 6 shows the split **FRKC4s** scheme is competitive with **ROCK4** [12]. However, in support of the splitting approach, it may be noted that **ROCK4** suffers significantly from internal stability issues arising from the finishing stages required for the nonlinear order conditions (discussed further in Section 4.2) which effectively limits the scheme to about 150 stages.

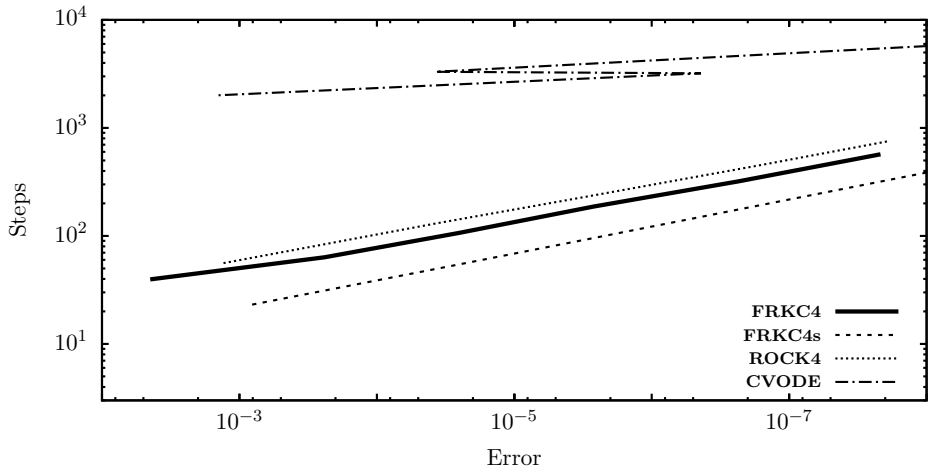


Figure 6: Performance results derived from the estimated error for the Brusselator problem. Shown are data for the fourth-order ESRK schemes **FRKC4s**, **FRKC6s**, **ROCK4**, and the fifth-order implicit **CVODE** scheme.

4.2. Butcher composition

An alternative to complex splitting is Butcher composition. At fourth-order, as illustrated in Table 1, $L - 4$ forward Euler steps are adopted from the **FRKC** stability polynomial with $N = 4$ as the scheme a, and appropriate finishing stages for the scheme $\overline{a}$ are subsequently derived.

According to a theorem of Hairer & Wanner [25], given the B-series a, b, the composite scheme ab is determined via

$$a\overline{a}(t) = \frac{1}{\alpha(t)} \sum \left(\sum_{i=0}^{q} \binom{q}{i} \overline{a}(s_i(t)) \prod_{z \in d_i(t)} a(z) \right), \tag{19}$$

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
c_2 & a_1 & & & & & \\
c_3 & a_1 & a_2 & & & & \\
c_4 & a_1 & a_2 & a_3 & & & \\
\vdots & \vdots & \vdots & \vdots & \ddots & & \\
c_L & a_1 & a_2 & a_3 & \cdots & a_{L-5} & \\
\hline
 & a_1 & a_2 & a_3 & \cdots & a_{L-5} & a_{L-4}
\end{array}
\qquad
\begin{array}{c|cccc}
0 & & & & \\
\overline{c}_2 & \overline{a}_{2\,1} & & & \\
\overline{c}_3 & \overline{a}_{3\,1} & \overline{a}_{3\,2} & & \\
\overline{c}_4 & \overline{a}_{4\,1} & \overline{a}_{4\,2} & \overline{a}_{4\,3} & \\
\hline
 & \overline{a}_1 & \overline{a}_2 & \overline{a}_3 & \overline{a}_4
\end{array}
$$

Table 1: **FRKC4** order composition tableaux.

where rooted trees t are used to represent derivatives in Taylor series. The first summation in Equation 19 is over all $\alpha(t)$ different labelings of t, $s_i(t)$ is the subtree formed by the first i indices, and $\{z \in d_i(t)\}$ is the difference set of subtrees formed by the remaining indices. The eight order conditions at fourth order are then given by

$$
\begin{aligned}
a\overline{a}(\bullet) &= a(\bullet) + \overline{a}(\bullet),\\
a\overline{a}(\bullet\!\!-\!\!\bullet) &= 2a(\bullet)\overline{a}(\bullet) + a(\bullet\!\!-\!\!\bullet) + \overline{a}(\bullet\!\!-\!\!\bullet),\\
a\overline{a}(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) &= 3a(\bullet)^2\overline{a}(\bullet) + 3a(\bullet)\overline{a}(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + \overline{a}(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet),\\
a\overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet) &= 3a(\bullet)\overline{a}(\bullet\!\!-\!\!\bullet) + 3a(\bullet\!\!-\!\!\bullet)\overline{a}(\bullet) + a(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet) + \overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet),\\
a\overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!-\!\!\bullet) &= 4a(\bullet)^3\overline{a}(\bullet) + 6a(\bullet)^2\overline{a}(\bullet\!\!-\!\!\bullet) + 4a(\bullet)\overline{a}(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!-\!\!\bullet) + \overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!-\!\!\bullet),\\
a\overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!>\!\!\bullet) &= 4a(\bullet)^2\overline{a}(\bullet\!\!-\!\!\bullet) + 4a(\bullet\!\!-\!\!\bullet)a(\bullet)\overline{a}(\bullet) + (8/3)a(\bullet)\overline{a}(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + (4/3)a(\bullet)\overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet),\\
&\quad + 2a(\bullet\!\!-\!\!\bullet)\overline{a}(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!>\!\!\bullet) + \overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!>\!\!\bullet),\\
a\overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!-\!\!\bullet\!\!>\!\!\bullet) &= 6a(\bullet)^2\overline{a}(\bullet\!\!-\!\!\bullet) + 4a(\bullet)\overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet) + 4a(\bullet\!\!-\!\!\bullet)\overline{a}(\bullet) + a(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!-\!\!\bullet\!\!>\!\!\bullet) + \overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!-\!\!\bullet\!\!>\!\!\bullet),\\
a\overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!>\!\!\bullet\!\!-\!\!\bullet) &= 4a(\bullet)\overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet) + 6a(\bullet\!\!-\!\!\bullet)\overline{a}(\bullet\!\!-\!\!\bullet) + 4a(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet)\overline{a}(\bullet) + a(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!>\!\!\bullet\!\!-\!\!\bullet) + \overline{a}(\bullet\!\!-\!\!\bullet\!\!<\!\!\bullet\!\!>\!\!\bullet\!\!-\!\!\bullet).
\end{aligned}
\tag{20}
$$

Hence, for given scheme a, imposing the required order conditions on $a\overline{a}$ yields equations for $\overline{a}$, which are in turn easily solved for $\overline{a}$. The reader is referred to [25] for further details.

Figure 6 shows a comparison of the **FRKC4** scheme based on composition methods with other schemes. The reference solution is provided by a fifth-order implicit preconditioned BDF solver from the **CVODE** numerical package [26]. The number of steps required for a given precision is comparable for **ROCK4** and **FRKC4** and somewhat greater than the split **FRKC4s** scheme. This difference may be ascribed to a loss of precision due to the accumulation of errors over the finishing stages [27].

5. Conclusions

FRKC extended stability Runge–Kutta (ESRK) schemes are shown to well-suited to the integration of large-scale problems governed by systems of PDEs where the explicit timescale is constrained by diffusion. An implementation of the **FRKC2** second-order schemes, publicly available at `maths.dit.ie/frkc`, is presented. The fourth-order **FRKC4** schemes are also presented with nonlinear order conditions addressed via both splitting and composition methods. These schemes are shown to be competitive with alternative ESRK methods.

5.1. Acknowledgments

I am grateful to organisers of Astronum2016 for the invitation to present this work in Monterey and to an anonymous referee for helpful comments.

6. References

[1] Saulev V 1960 *Fizmatgiz, Moscow*
[2] Guillou A and Lago B 1960 *Ier Congr. Ass. Fran. Calcul., AFCAL* 43–56
[3] Gentzsch W and Schluter A 1978 *Z. Angew. Math. Mech* **58** T415–T416
[4] Alexiades V, Amiez G and Gremaud P 1996 *Communications in numerical methods in engineering* **12** 31–42
[5] O'Sullivan S and Downes T P 2006 *Monthly Notices of the Royal Astronomical Society* **366** 1329–1336
[6] O'Sullivan S and Downes T P 2007 *Monthly Notices of the Royal Astronomical Society* **376** 1648–1658
[7] Medovikov A A 1998 *BIT Numerical Mathematics* **38** 372–390
[8] van Der Houwen P J and Sommeijer B P 1980 *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift fr Angewandte Mathematik und Mechanik* **60** 479–485
[9] Verwer J G 1996 *Applied Numerical Mathematics* **22** 359–379
[10] Sommeijer B, Shampine L and Verwer J 1998 *Journal of Computational and Applied Mathematics* **88** 315–326
[11] Abdulle A and Medovikov A A 2001 *Numerische Mathematik* **90** 1–18
[12] Abdulle A 2002 *SIAM Journal on Scientific Computing* **23** 2041–2054
[13] Martin-Vaquero J and Janssen B 2009 *Computer Physics Communications* **180** 1802–1810
[14] Meyer C D, Balsara D S and Aslam T D 2012 *Monthly Notices of the Royal Astronomical Society* **422** 2102–2115
[15] Meyer C D, Balsara D S and Aslam T D 2014 *Journal of Computational Physics* **257** 594–626
[16] O'Sullivan S 2015 *Journal of Computational Physics* **300** 665 – 678
[17] Van Der Houwen P J 1977 *Construction of integration formulas for initial value problems* (North Holland)
[18] Lebedev V I 2000 *Computational mathematics and mathematical physics* **40** 1729–1740
[19] Lebedev V 1994 *Numerical methods and applications* 45–80
[20] Hairer E and Wanner G 1996 *Springer series in computational mathematics* **14**
[21] Nicolis G and Prigogine I 1977 *Self-organization in nonequilibrium systems : from dissipative structures to order through fluctuations* (New York: Wiley)
[22] Abdulle A 2001 *Chebyshev methods based on orthogonal polynomials* Ph.D. thesis
[23] Castella F, Chartier P, Descombes S and Vilmart G 2009 *BIT Numerical Mathematics* **49**(3) 487–508
[24] Blanes S, Casas F, Chartier P and Murua A 2013 *Mathematics of Computation* **82** 1559–1576
[25] Hairer E, Nørsett S and Wanner G 1993 *Springer series in computational mathematics* **8**
[26] Cohen S D and Hindmarsh A C 1996 *Computers in physics* **10** 138–143
[27] Hundsdorfer W and Verwer J G 2013 *Numerical solution of time-dependent advection-diffusion-reaction equations* vol 33 (Springer Science & Business Media)