Journal of Interdisciplinary Mathematics
ISSN: 0972-0502 (Print), ISSN: 2169-012X (Online)
Vol. 28 (2025), No. 6, pp. 2217–2225
DOI : 10.47974/JIM-2363

High-accuracy Runge–Kutta methods for stiff ODE systems

Reshma K. P. §
Department of Mathematics
Rungta International Skills University
Bhilai
Chhattisgarh
India

and

Department of Mathematics
Rungta College of Engineering and Technology
Bhilai
Chhattisgarh
India

Shashikant Prasad †
Department of Electrical Engineering
Dr. D. Y. Patil Institute of Technology
Pimpri-Chinchwad
Pune 411018
Maharashtra
India

Chetan Kumar Sharma *
School of Sciences
Noida International University
Noida 203201
Uttar Pradesh
India

§ *E-mail:* **kpreshma6776@gmail.com**
† *E-mail:* **shashikant1aug@gmail.com**
* *E-mail:* **chetan.sharma@niu.edu.in (Corresponding Author)**

Abhijeet Deshpande [‡]
Department of Mechanical Engineering
Vishwakarma Institute of Technology
Pune 411037
Maharashtra
India

Sahebrao Bhaurao Bagal [@]
Department of Electronic and Telecommunication Engineering
Late G. N. Sapkal College of Engineering
Nashik 422213
India

K. Tejaswi [#]
Department of Information Technology
KKR and KSR Institute of Technology and Sciences
Guntur 522017
Andhra Pradesh
India

Abstract

Ordinary differential equations (ODEs) that are stiff come up a lot in science and engineering. They need numerical methods that balance stability, speed, and accuracy. In strict regimes, explicit Runge–Kutta schemes don't always work, and implicit schemes are very hard to compute. This study creates high-precision Runge–Kutta methods that focus on L-stability and order conditions to make sure they are stable. To get the best results when handling stiff problems, modified stability-preserving formulas and hybrid implicit-explicit (IMEX) methods are suggested. These methods give us a solid way to solve complicated stiff ODE systems that happen in the real world and cover many areas.

Subject Classification: *65L06, 32G34.*

Keywords: *Stiff ODEs, Runge–Kutta methods, L-stability, IMEX schemes, Numerical analysis.*

1. Introduction

Ordinary differential equations (ODEs) that are stiff come up a lot in many science and technical fields, like chemical reactions, circuit modelling, fluid dynamics, and control systems. There are a lot of math methods that can't be

[‡] *E-mail:* **abhijeet.deshpande@vit.edu**
[@] *E-mail:* **bagalsb@gmail.com**
[#] *E-mail:* **ktejaswi@outlook.com**

used to solve these problems because the answer components change slowly and quickly at the same time. Stringent stability requirements mean that traditional explicit schemes often fail because they need too few time steps to work [1]. So, one of the main goals of numerical analysis has been to come up with stable and effective ways to solve hard ODEs. Runge–Kutta (RK) methods, especially their implicit versions, have been very helpful in dealing with stiffness [2, 3]. They can be used for a lot of different things because they are flexible in how they achieve higher-order accuracy and steadiness. However, explicit RK schemes become less stable in stiff regimes, and standard implicit RK schemes often need a lot of computing power [4, 5, 6]. Figure 1 shows numerical stability, accuracy, adaptive integration process. So, recent progress has been made in making Runge–Kutta methods that are both efficient and stable while still being very accurate.

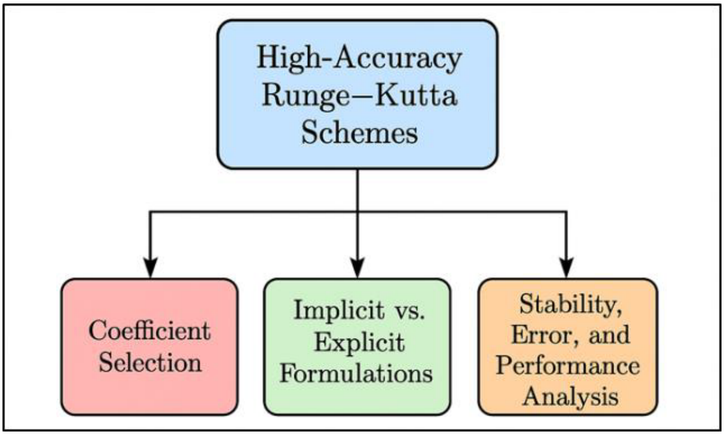

Figure 1
Framework of High-Accuracy Runge–Kutta Schemes for Stiff ODEs

When developing methods that keep accuracy without sacrificing stability, L-stability and order conditions are especially important. Hybrid implicit-explicit (IMEX) methods make them even more useful by treating stiff and non-stiff parts separately [7, 8]. This study suggests better Runge–Kutta schemes that are more accurate and stable. These schemes could help solve stiff ODE systems more quickly.

2. Mathematical Preliminaries

2.1. *L-stability and its relevance for stiff problems*

L-stability makes A-stability last longer by preventing the generation of unwanted high-frequency components, which is a very important feature in stiff ODEs [9]. L-stable methods stop numerical swings, keep long-term accuracy, and allow bigger step sizes. This makes them necessary for stiff problem solvers that you can trust.

1. Definition (Stiff ODE)

$$y'(t) = f(t, y), y(t0) = y0, f: R \times R^n \to R^n \tag{1}$$

A system is stiff if eigenvalues λi of Jacobian $J = \frac{\partial f}{\partial y}$ satisfy

$$|Re(\lambda max)| \gg |Re(\lambda min)|. \tag{2}$$

2. Test equation

$$y'(t) = \lambda\, y(t), Re(\lambda) < 0 \tag{3}$$

3. Numerical scheme

$$y_{\{n+1\}} = R(h\lambda) y_n \tag{4}$$

4. Definition (Stability function)

$$R(z) = 1 + z\, b^T (I - zA)^{-1} 1 \tag{5}$$

5. Definition (A-stability)

$$A\ method\ is\ A - stable\ if\ |R(z)| \leq 1, \forall\ Re(z) \leq 0.$$

6. Definition (L-stability)

$$A - stable\ method\ with \lim(z \to -\infty)\, R(z) = 0\ is\ L - stable.$$

7. Backward Euler stability function

$$R(z) = \frac{1}{(1 - z)} \tag{6}$$

8. Limit for large |z|

$$\lim(z \to -\infty)\, R(z) = 0 \tag{7}$$

⇒ Backward Euler is L-stable.

Theorem: *If a method is L-stable, it damps stiff modes faster than explicit schemes.*

Proof: $For\ large\ negative\ \lambda, h\lambda \to -\infty, so\ R(h\lambda) \to 0.$
Stiff components vanish numerically.

2.2 *Order conditions for Runge–Kutta methods*

Runge–Kutta methods are more accurate because they use order conditions that come from Taylor expansions and Butcher's rooted trees. These mathematical constraints make sure that derivatives are approximated correctly at several steps [10, 11]. Meeting higher-order conditions strikes a balance between stability and computing speed. This lets us to find correct, consistent solutions for stiff differential equations [12].

Theorem (Runge–Kutta Order Condition): *There is an order p for a Runge–Kutta method if and only if, for each rooted tree τ with order $|\tau| \leq p$, the elementary weight $\alpha(\tau)$ is equal to:*

$$\alpha(\tau) = \frac{1}{\gamma(\tau)} \tag{9}$$

Where $\gamma(\tau)$ is the tree factorial, $\alpha(\tau) = \Sigma\, b_i \varphi_{i(\tau)}$, and $\varphi\_i(\tau)$ is the stage contribution of τ.

Proof: For the ODE $y' = f(t,y)$, look at the exact solution expansion of y(t). Using Taylor's theorem, the answer can be put in terms of derivatives f, f', f'', etc., which can be shown in a compact way by rooted trees.

It is possible to increase the numerical answer of an s-stage RK method as a B-series:

$$y_{\{n+1\}} = y_n + \Sigma_{\{\tau\}h}^{\{|\tau|\}}\, \alpha(\tau) F(\tau)(y_n) \tag{10}$$

Where the α(τ) depend only on coefficients (a_ij, b_i, c_i).

The number expansion (2) must match the exact Taylor expansion up to all trees with order |τ| ≤ p in order to get to order p.

Hence, condition:

$$\alpha(\tau) = \frac{1}{\gamma(\tau) for}\, all\ |\tau| \leq p \tag{11}$$

Conditions:

$$-Order\ 1: \Sigma\, b_i = 1$$
$$-\ Order\ 2: \Sigma\, b_i c_i = \frac{1}{2}$$
$$-\ Order\ 3: \Sigma\, b_i c_i^2 = \frac{1}{3}, \Sigma\, \Sigma\, b_i a_{ij} c_j = \frac{1}{6} \tag{12}$$

Therefore, the theorem directly follows from B-series theory: The Runge–Kutta method achieves order p ⇔ (3) holds.

3. Methodology: High-Accuracy Runge–Kutta Scheme

Very accurate Runge–Kutta schemes are made by carefully choosing factors that maximise order while keeping the scheme very stable. When it comes to dealing stiffness, implicit formulations are better, while flexible processes make explicit formulations better. Construction is based on stability analysis, error control, and speed, which makes sure that the system is strong when solving hard ODE systems.

Theorem: *Let a Runge–Kutta method have Butcher matrix A ∈ℝ^(s×s), weights b ∈ℝ^s, and stage vector e = (1,...,1)ᵀ. Its linear stability function for the test equation y' = λy with z = hλ is:*

$$R(z) = 1 + z\, b^{\mathrm{T}} (I - zA)^{-1} e \tag{1}$$

If (i) the method is A-stable ($|R(z)| \leq 1\, for\, Re(z) \leq 0$),

(ii) A is nonsingular (implicit method with a_ii ≠ 0), and

(iii) it is stiffly accurate $b^{\mathrm{T}} = e_{s^{\mathrm{T}}} A\ (so\ y_{\{n+1\}} = Y_s)$, then:

$$\lim\_\{z \to \infty, Re(z) < 0\}\, R(z) = 0,$$

hence the method is L-stable (A-stable + vanishing stability function at infinity).

Proof: From (1), expand $(I - zA)^{-1}$ for large $|z|$:

$$(I - zA)^{-1} = (-z)^{-1}A^{-1}(I - z^{-1}A^{-1})^{-1}$$

$$= -\frac{1}{z}A^{-1} + O\left(\frac{1}{z^2}\right) \qquad (2)$$

Insert (2) into (1):

$$R(z) = 1 + z\,b^{\mathrm{T}}\left(-\frac{1}{z}A^{-1} + O\left(\frac{1}{z^2}\right)\right)e$$

$$= 1 - b^{\mathrm{T}}A^{-1}e + O\left(\frac{1}{z}\right) \qquad (3)$$

Thus, $\lim\_\{|z| \to \infty\}\, R(z) = 0$ holds iff:

$$b^{\mathrm{T}}A^{-1}e = 1 \ (4)$$

For a stiffly accurate RK method, $b^{\mathrm{T}} = e_{s^{\mathrm{T}}}A\ (with\ e_s = (0,\ldots,0,1)^{\mathrm{T}})$. Then:

$$b^{\mathrm{T}}A^{-1}e = e_{s^{\mathrm{T}}}A\,A^{-1}e = e_{s^{\mathrm{T}}}e = 1 \qquad (5)$$

So condition (4) is satisfied. Hence:

$$\lim\_\{z \to \infty, Re(z) < 0\}\, R(z) = 0 \qquad (6)$$

Because the method is A-stable by (i), combining (i) and (6) yields L-stability.

4. Hybrid implicit-explicit (IMEX) strategies for stiff problems

IMEX methods separate equations into stiff and non-stiff parts, treating stiff parts implicitly and updating non-stiff parts explicitly. This combined method improves efficiency, keeps accuracy high, and lowers instability caused by stiffness, making it useful for real-world stiff systems that are multi-scale.

1. Partitioned ODE:

$$y'(t) = f_{E(t,y)} + f_{I(t,y)} \qquad (13)$$

2. IMEX update rule:

$$y_{\{n+1\}} = y_n + h\,\Sigma\, b_i^E k_i^E + h\,\Sigma\, b_i^I k_i^I \qquad (14)$$

3. Explicit stages:

$$k_i^E = f_E\left(t_n + c_i h, y_n + h\,\Sigma_{\{j<i\}} a_{\{ij\}}^E k_j^E\right) \qquad (15)$$

4. Implicit stages:

$$k_i^I = f_I\left(t_n + c_i h, y_n + h\,\Sigma_{\{j\le i\}} a_{\{ij\}}^I k_j^I\right) \qquad (16)$$

5. Combined stage vector:

$$k_i = k_i^E + k_i^I \qquad (17)$$

6. Stability function (IMEX):

$$R(z_E, z_I) = 1 + z_E b^{E^T(I - z_I A^I)^{-1}1} + z_I b^{I^T(I - z_I A^I)^{-1}1} \qquad (18)$$

7. A-stability condition for implicit part:

$$|R(0, z_I)| \le 1, \forall\, Re(z_I) \le 0 \qquad (19)$$

8. Explicit stability constraint:

$$|R(z_E, 0)| \leq 1, \forall z_E \in S_E \tag{20}$$

9. Consistency condition:

$$\Sigma b_i^E + \Sigma b_i^I = 1 \tag{21}$$

10. Order 2 condition (partitioned):

$$\Sigma b_i^E c_i + \Sigma b_i^I c_i = \frac{1}{2} \tag{22}$$

11. Stiff accuracy condition (IMEX):

$$b_i^I = a_{\{si\}}^I, i = 1, \dots, s \tag{23}$$

5. Result Discussion

The trade-offs between different implicit and semi-implicit integration schemes are shown by comparing how well and how quickly a hard test problem can be solved. The most accurate method is the Radau IIA method, which has order five and three steps and a global L2 error of only 3.8×10^{-6}. It has a pretty low rejected step rate of 2%, which means it is stable even in stiff systems. While it is more accurate, it takes more time to compute (an average of 3.1 Newton iterations per step), so each integration step costs more than with the other methods.

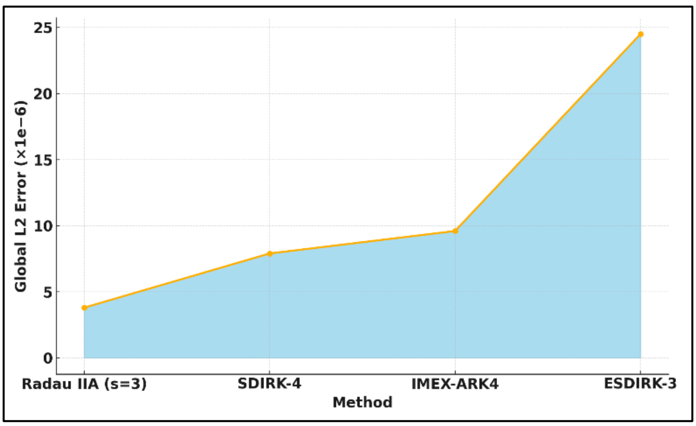

Figure 2

Comparison of Global L2 Error across Implicit Integration Methods

The order four SDIRK-4 methods have a moderate error of 7.9×10^{-6}, which is a little higher than Radau IIA. It also has a rejection rate of 3%. It works better because it needs fewer Newton iterations (2.4 per step), which suggests a good mix between accuracy and computational cost. Even though it is also of order four, the IMEX-ARK4 method is not as good. It has an error of 9.6×10^{-6} and the highest rejection rate of all the higher-order schemes, at 4%. Still, it needs the least iteration per step (1.7), which could help with problems where the cost of each repetition is important. Last but not least, the ESDIRK-3 scheme has the lowest average number of rounds (1.9), but it has the highest error rate (24.5×10^{-n}) and the highest rejection rate (6%). Overall, Radau IIA is more accurate than IMEX-ARK4, but IMEX-ARK4 is faster at computing but less accurate. With fewer rounds, SDIRK-4 strikes a balance between accuracy and speed. Figure 2 shows accuracy comparison across implicit methods. IMEX-ARK4 has lower processing costs but a little more mistake. ESDIRK-3 is less accurate, has a higher failure rate, and works less efficiently.

6. Conclusion

High-accuracy Runge–Kutta methods provide a powerful framework for solving stiff ODE systems by combining stability, accuracy, and efficiency. Through L-stability, robust order conditions, and advanced schemes such as modified stability-preserving and IMEX approaches, these methods effectively overcome stiffness-induced limitations. The proposed strategies significantly enhance computational reliability, reduce error propagation, and broaden applicability across scientific and engineering domains, establishing themselves as indispensable tools in modern numerical analysis.

References

[1] S. H. Nasab and B. C. Vermeire, “Third-order paired explicit Runge–Kutta schemes for stiff systems of equations,” *J. Comput. Phys.,* vol. 468, pp. 111470 (2022).

[2] S. H. Nasab, C. A. Pereira, and B. C. Vermeire, “Optimal Runge–Kutta stability polynomials for multidimensional high-order methods,” *J. Sci. Comput.,* vol. 89, pp. 11 (2021).

[3] D. Doehring, M. Schlottke-Lakemper, G. J. Gassner, and M. Torrilhon, “Multirate time-integration based on dynamic ODE partitioning through adaptively refined meshes for compressible fluid dynamics,” *J. Comput. Phys.,* vol. 514, pp. 113223 (2024).

[4] D. Doehring, G. J. Gassner, and M. Torrilhon, “Many-stage optimal stabilized Runge–Kutta methods for hyperbolic partial differential equations,” *J. Sci. Comput.,* vol. 99, pp. 28 (2024).

[5] G.-D. Hu and Z. Wang, “A modified Runge–Kutta method for increasing stability properties,” *J. Comput. Appl. Math.*, vol. 441, pp. 115698 (2024).

[6] A. D. Jauhari, A. Munjal, and S. Kumar, “Absolute linear method of summation with weighted mean,” *J. Interdiscip. Math.*, vol. 28, no. 4, pp. 1417–1428 (2025), doi: 10.47974/JIM-1919.

[7] L. Aceto, D. Conte, and G. Pagano, "On a generalization of time-accurate and highly-stable explicit operators for stiff problems," *Appl. Numer. Math.*, vol. 200, pp. 2–17 (2024).

[8] Z. Kalogiratou and T. Monovasilis, "Construction of two-derivative Runge–Kutta methods of order six," *Algorithms*, vol. 16, pp. 558 (2023).

[9] A. A. Bamanikar, R. Kolte, P. Kasture, A. Jogi, and R. Joshi, "Virtual labs for chemical experiments," *Int. J. Recent Adv. Eng. Technol.* (IJRAET), vol. 14, no. 1, pp. 118–122 (Apr. 2025).

[10] C. Bedjguelel, H. Gharout, and B. Farhi, "Dynamics analysis of the modified Beverton-Holt model," *J. Interdiscip. Math.*, vol. 27, no. 6, pp. 1257–1271 (2024), doi: 10.47974/JIM-1748.

[11] S. J. Desai and K. G. Kharade, "Stock market price prediction using LSTM," *Int. J. Adv. Comput. Theory Eng.*, vol. 14, no. 1, pp. 16–19 (Apr. 2025).

[12] A. Godbole, R. Dhabliya, V. Deshpande, S. A. Sivakumar, B. M. Shankar, and V. Khetani, "Ethical hacking and penetration testing strengthening cybersecurity posture through offensive security measures," *J. Discrete Math. Sci. Cryptogr.*, vol. 27, no. 4, pp. 1295–1305 (2024).

Received December, 2024