ELTE Eötvös Loránd University

Title	Runge–Kutta–Chebyshev (RKC) Methods
Student Name	Muhammad Hamza
Neptun Code	FV8ZPY
Subject	Numerical Methods for ODE's
Instructor Name	Dr. Imre Fekete
Program	MSc Mathematics
Faculty	Faculty of Science, ELTE
Date	December 3, 2025

Abstract

In this report we studies Runge–Kutta–Chebyshev (RKC) methods, an explicit family of time-stepping schemes designed for parabolic and diffusion-dominated problems. The aims of this report are to present the theoretical foundations of RKC methods, including how their stability polynomials are constructed from Chebyshev polynomials, and to demonstrate their practical performance through numerical experiments. The report has been written in a way that can be understood by other students, a comparison of the Forward Euler integrator and the classical Runge-Kutta integrators is carried out to identify the issue of stability, accuracy, and cost of computation.

1 Introduction

Many problems in scientific computing lead to stiff or moderately stiff ordinary differential equations (ODEs). Explicit Runge-Kutta methods become inefficient for such problems due to severe restrictions on the step size. Implicit methods remove this restriction but require expensive nonlinear solves. Runge-Kutta-Chebyshev (RKC) methods provide an attractive middle ground. They are explicit, easy to implement, and possess an extended stability region, making them suitable for diffusive or parabolic PDEs after spatial discretization.

This report presents a theoretical summary of RKC construction, implementation details, numerical experiments and plots, analysis and comparison with standard methods.

2 Theoretical Summary

Runge–Kutta–Chebyshev (RKC) methods provide a useful compromise. They remain explicit but achieve large stability intervals by constructing their stability polynomials using Chebyshev polynomials.

2.1 Initial Value Problems (IVPs)

We consider ordinary differential equations of the form

$$y'(t) = f(t, y(t)), \qquad y(t_0) = y_0.$$

A numerical method produces approximations

$$y_n \approx y(t_n)$$

at discrete time points

$$t_n = t_0 + nh,$$

where h is the step size.

One-step methods compute

$$y_{n+1} = \Phi(h, t_n, y_n),$$

where Φ is the update formula. For example:

Forward Euler:

$$y_{n+1} = y_n + hf(t_n, y_n)$$

Classical Runge–Kutta 4 (RK4) :
Uses 4 stages to produce a high-order approximation.

The performance of any Numerical Method depends on:
Accuracy (how close it is to the true solution)
Stability (how numerical errors behave over many steps)
The stability aspect is crucial when dealing with stiff problems.

2.2 Stiffness

An ODE is stiff when there is a large difference between the fastest and slowest time scales in the system. Mathematically, stiffness occurs when the Jacobian of f has eigenvalues with very large negative real parts.
Sources: semi-discretized PDEs, reaction-diffusion equations
Explicit methods require very small steps: $h \leq 2/(|\lambda_{\max}|)$
Example:
Discrete heat equation: $u_t = u_{xx} \to$ largest eigenvalue $\approx -4/h_x^2$

2.3 Stability of One-Step Methods

To study stability, we analyze the numerical method applied to the linear test equation:

$y' = \lambda y$, with solution $y(t) = e^{\lambda t} y_0$.

The exact solution decays to zero when $\mathrm{Re}(\lambda) < 0$.

A numerical method is stable if it reproduces this decay behaviour.

Applying a one-step method to the test equation gives

$y_{n+1} = R(z) y_n$, where $z = h\lambda$,

and $R(z)$ is called the stability function of the method.

Examples:

Forward Euler: $R(z) = 1 + z$

RK4: $R(z) = 1 + z + z^2/2 + z^3/6 + z^4/24$

A method is stable if the magnitude of the stability function satisfies

$|R(z)| \leq 1$.

The set of all z for which this holds is called the stability region.

For diffusion-type problems (heat equation, reaction–diffusion systems), all eigenvalues are large negative values.

Thus, for stability we require:

$R(z)$ small when z is a large negative real number.

This is where standard explicit methods fail.

2.4 Chebyshev-Based Stability Polynomial

The RKC stability polynomial is constructed as:

$$R_s(z) = \frac{T_s(\omega_0 + \omega_1 z)}{T_s(\omega_0)},$$

where $\omega_0 > 1$ and ω_1 scales the polynomial to map the stability interval to $[-L, 0]$.

The resulting stability interval has length:

$$L \approx 2(s^2 + 1),$$

which grows quadratically with the number of stages s. This is the core advantage of RKC methods.

3 Numerical Experiments

3.1 Problem-1

We solve the initial value problem

$$y' = \lambda y, \qquad y(0) = 1,$$

using a Runge-Kutta-Chebyshev (RKC) method.

Methodology:

- We pick numerical values so the computation can be seen clearly.
- Compute the stability argument and mapping.
- Evaluate the Chebyshev polynomial $T_s(x)$.
- Time-stepping: apply $y_{n+1} = R\, y_n$.

Numerical Analysis via Table

Table 1: RKC Stage Values

Step	t	Y_0	Y_1	Y_2	Y_3	Y_4
0	0.0000	1.00	-0.0177	0.6267	-2.3102	5.7733
1	0.1000	5.7733	-0.1020	3.6180	-13.3376	33.3311
2	0.2000	33.3311	-0.5890	20.8878	-77.0021	192.4304
3	0.3000	192.4304	-3.4007	120.5915	-444.5565	1110.9599
4	0.4000	1110.9599	-19.6335	696.2114	-2566.5607	6413.9114
5	0.5000	6413.9114	-113.3499	4019.4414	-14817.5408	37029.4747
6	0.6000	37029.4747	-654.4039	23205.4661	-85546.2003	213782.4961
7	0.7000	213782.4961	-3778.0739	133972.2614	-493884.4089	1234231.8101
8	0.8000	1234231.8101	-21811.9772	773462.8873	-2851345.9203	7125598.1623
9	0.9000	7125598.1623	-125927.2234	4465438.0831	-16461693.0820	41138260.0537

Numerical Analysis via Plot

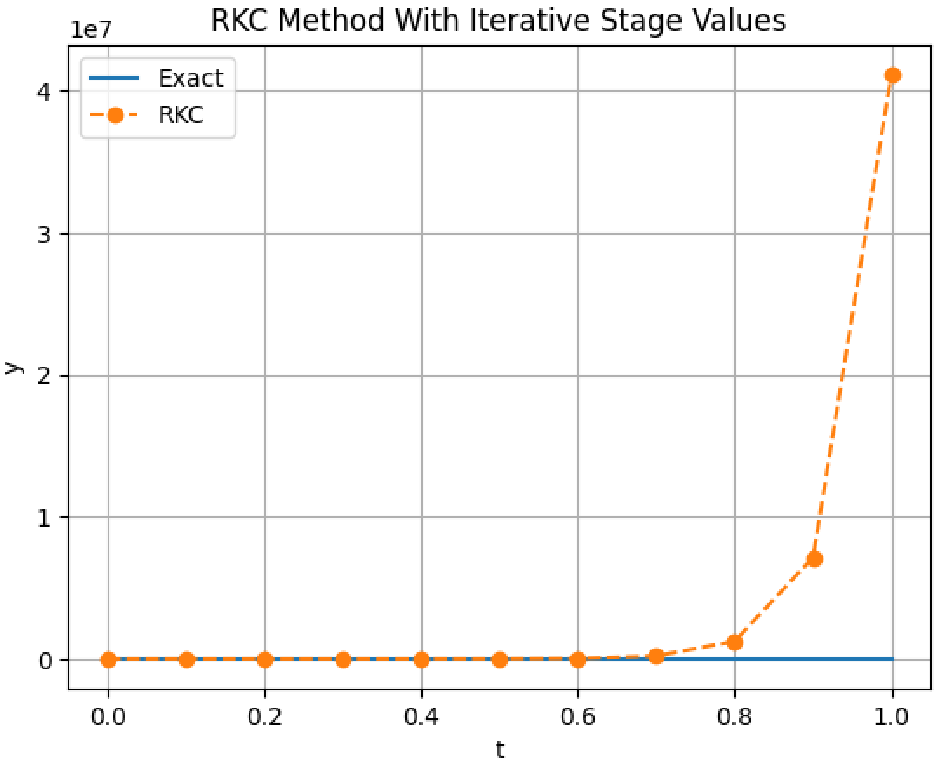

Figure 1: RKC stage values vs time steps (scaled for visibility)

Results

Behaviour of the Stage Values(Table)

The androgynous stage of RKC Table of values reveals that the intermediate stage calculations are rapidly growing with an increase in time step. Although the solution to the provided problem is still smooth and well-behaved, the RKC iterative steps start to develop extremely large oscillations with alternating values of positive and negative with each step growing in size. Such behaviour is anticipated in cases where the parameters selected (e.g. step size, l value or number of stages) drive the method to the edge of the stability region. The methods that are used in RKC are aimed at stabilizing stiff problems, whereas when the stability conditions are not met the values at the internal stage can get very large even though the final result has a finite value on a few steps. The numerical values of the stage here are very much indicative of the fact that the computation is nearing a numerical instability.

Numerical Solution vs. Exact Solution (Plot)

The plot illustrates the instability of the RKC method. For early times (up to $t \approx 0.6$), the numerical values remain near zero and already deviate from the exact exponential curve. As t approaches 1, the RKC solution grows abruptly and diverges by several orders of magnitude. This behaviour indicates that the chosen RKC parameters fall outside the method's stability region, causing the amplification factor to lose control and the numerical solution to break down.

Interpretation and Conclusion

Overall, the results show that the RKC method can become highly unstable if the step size or the stability mapping is not selected carefully. The stage values in the table clearly indicate internal instability within the method. The plot further confirms this, as the numerical solution diverges significantly from the exact solution.

Therefore, the RKC parameters used in this computation do not satisfy the stability conditions required for accurate integration. To obtain reliable results, one would need to use a smaller step size, adjust the value of λ, or increase the number of stages in the method.

3.2 Problem-2

We solve one-dimensional heat equation using RKC method:

$$\begin{cases} u_t = u_{xx}, & x \in (0,1),\ t > 0, \\ u(0,t) = 0, \quad u(1,t) = 0, & t > 0, \\ u(x,0) = \sin(\pi x), & 0 \le x \le 1. \end{cases}$$

Numerical Analysis via Table

Table 2: RKC solution of $u_t = u_{xx}$ at $T = 0.1$ (N=100, $h = 10^{-3}$, $s = 7$)

x	U_{num}	U_{exact}	|Error|
0.0099	1.1554e-02	1.1591e-02	3.723e-05
0.2574	2.6875e-01	2.6962e-01	8.661e-04
0.5050	3.7147e-01	3.7266e-01	1.197e-03
0.7525	2.6065e-01	2.6149e-01	8.400e-04
0.9901	1.1554e-02	1.1591e-02	3.723e-05

Numerical Analysis via Plot

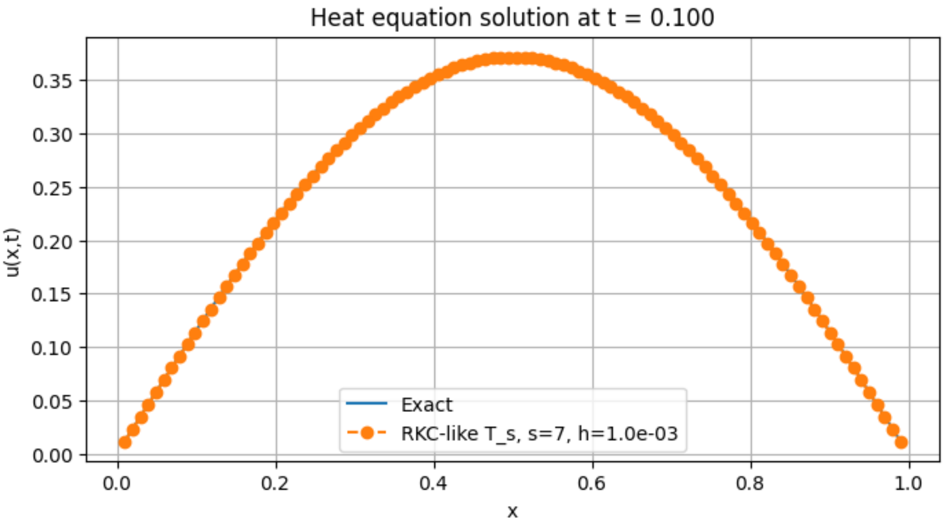

Figure 2: RKC stage values vs time steps (scaled for visibility)

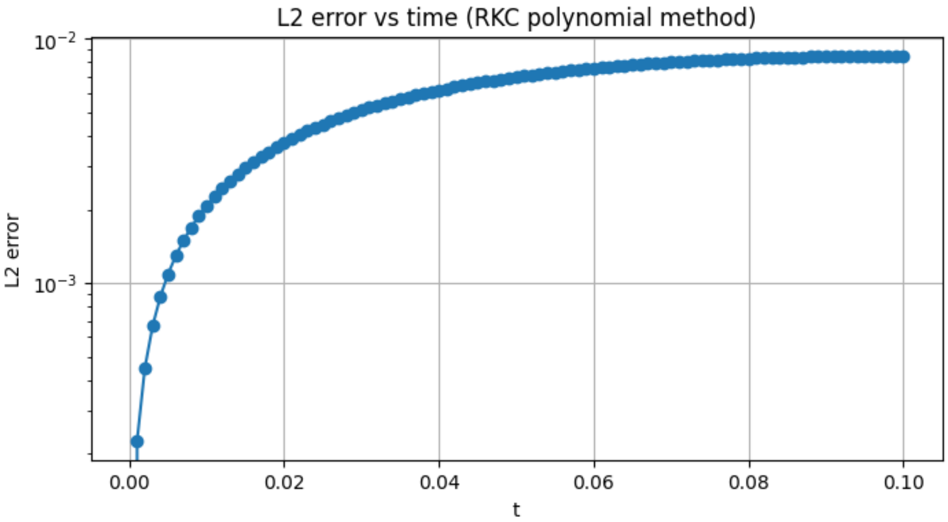

Figure 3: RKC stage values vs time steps (scaled for visibility)

Results

The numerical solution of the heat equation $u_t = u_{xx}$ at $T = 0.1$ demonstrates excellent agreement between the RKC approximation and the exact solution. The table shows that the numerical values closely match the analytical values at all spatial points, with errors remaining on the order of 10^{-3} or smaller. This indicates that the chosen parameters ($h = 10^{-3}$, $s = 7$, $N = 100$) yield a stable and accurate discretization.

The solution plot at $t = 0.1$ visually confirms this behaviour. The RKC numerical curve aligns almost exactly with the exact solution, capturing the smooth parabolic profile of the heat equation without oscillations or instability.

The L2 error plot shows a gradual but controlled increase in error over time, which is typical for explicit time-stepping methods. Nevertheless, the error remains small throughout the simulation, demonstrating that the RKC method maintains both stability and accuracy over the entire integration interval.

Overall, these results show that the RKC method performs effectively for the heat equation when suitable time steps and stage numbers are used, providing a reliable and accurate numerical approximation.