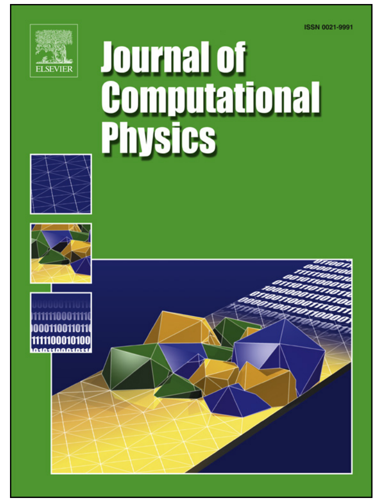
Super-time-stepping schemes for parabolic equations with boundary conditions

Skaras, Timothy G.
Saxton, Torrey A.
Meyer, Chad D.
Aslam, Tariq Dennis

To be published in: Journal of Computational Physics

DOI to publisher's version: 10.1016/j.jcp.2020.109879

Journal Pre-proof



Super-time-stepping schemes for parabolic equations with boundary conditions

Timothy Skaras, Torrey Saxton, Chad Meyer and Tariq D. Aslam

PII: S0021-9991(20)30653-7
DOI: https://doi.org/10.1016/j.jcp.2020.109879
Reference: YJCPH 109879

To appear in: *Journal of Computational Physics*

Received date: 2 February 2020
Revised date: 25 September 2020
Accepted date: 26 September 2020

Please cite this article as: T. Skaras, T. Saxton, C. Meyer et al., Super-time-stepping schemes for parabolic equations with boundary conditions, *Journal of Computational Physics*, 109879, doi: https://doi.org/10.1016/j.jcp.2020.109879.

Journal of Computational Physics (2020)

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Super-time-stepping schemes for parabolic equations with boundary conditions

Timothy Skaras[a], Torrey Saxton[b], Chad Meyer[c], Tariq D. Aslam[d]

[a] *Merton College, Oxford, OX1 4JD, UK*
[b] *Department of Physics, University of Notre Dame, Notre Dame, IN 46556 USA*
[c] *Computational Division, XCP-4, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*
[d] *Theoretical Division, Physics and Chemistry of Materials (T-1), Los Alamos National Laboratory, Los Alamos, NM 87545, USA*

ARTICLE INFO

Article history:

ABSTRACT

We present a super-time-stepping scheme for numerically solving parabolic partial differential equations with Dirichlet boundary conditions (BC). Using the general Forward Euler scheme, one can show that by taking varying step sizes there is the potential of propagating the solution forward in time by a greater amount than with uniform step sizes, while maintaining the same order of accuracy. As shown in [1] and [2], if one further requires that the scheme have the Convex Monotone Property (CMP), then there exists a scheme which results in linear, monotone stability of the solution. This monotone stability is highly desirable in many physical situations, such as thermal diffusion, where the physical system will not oscillate, but will behave monotonically. However, the schemes devised in [3], [4], [1], and [2] do not include situations that have a boundary condition, and the inclusion of boundary conditions will henceforth be our focus. It is shown that a particular Runge-Kutta-Gegenbauer class of schemes [5] will maintain the CMP even in the presence of Dirichlet BC.

1. Introduction

There are many physical situations which can be described by a parabolic partial differential equation (PDE), whereas there are not many numerical methods designed to handle these problems in an efficient manner, requiring egregious amounts of computational resources for accurate answers. Physical situations which include a boundary condition are of particular interest — thermal diffusion in a system with boundaries held at a fixed temperature, for example. This situation is still described by the heat equation, but the boundary condition causes many issues in terms of analytic and numerical solutions for the problem. Here, we provide the Runge-Kutta Gegenbauer (RKG) scheme, as presented by [5], which is a super-time-stepping (STS) scheme that solves this boundary condition problem and has the Convex Monotone Property (CMP) as defined in [1].

Suppose we are solving a scalar, parabolic PDE with solution $u(x,t)$. We discretize the x-t plane with spatial grid points $x_0, x_1, \ldots$ separated by spatial step Δx and temporal grid points $t_0, t_1, \ldots$ separated by time step Δt. Let u_i^n denote a numerical approximation of $u(x_i, t_n)$ where x_i is some spatial grid point and t_n is some temporal grid point. This numerical approximation will be determined by the numerical scheme we choose and the initial condition

$$u_i^0 = u(x_i, t_0). \tag{1}$$

Let u^n denote the vector of approximations u_i^n at temporal step t_n. For our purposes, our numerical solution u_i^{n+1} at spatial grid point x_i will be some function of u^n and i. In other words,

$$u_i^{n+1} = \mathcal{H}\left(u^n; i\right). \tag{2}$$

When solving such a PDE, it is useful for the method to be monotone stable. We say that a numerical method is monotone or monotone stable if at any time step n

$$v_j^n \geq u_j^n \quad \forall j \quad \implies \quad v_j^{n+1} \geq u_j^{n+1} \quad \forall j, \tag{3}$$

where v_j^n and u_j^n are the numerical solutions resulting from any two initial conditions. In words, this property guarantees that if one solution v is greater than or equal to u at every point in space, then this will remain true for all times. It has been shown by [6] that

$$\frac{\partial}{\partial u_i^n}\mathcal{H}\left(u^n; j\right) \geq 0 \quad \text{for all } i, j, u^n \tag{4}$$

is a sufficient condition for a method to be monotone. By imposing Dirichlet Boundary conditions, we then follow the derivation in section 3 of [1] for the point immediately adjacent to the boundary, as this point will be most affected by the boundary, to derive a new super-time-stepping scheme.

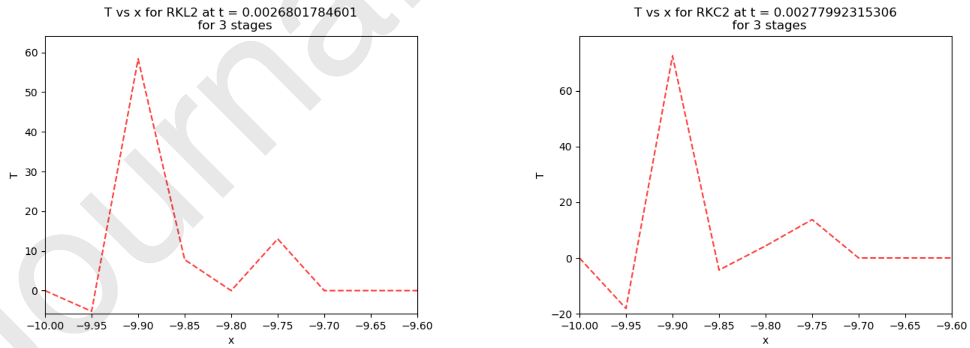


Fig. 1
Using RKL2 (Left) and RKC2 (Right), a delta function of T = 100 °C at $x = -9.9$ cm with T = 0 °C elsewhere and a boundary condition of T($x = -10$ cm) = 0 °C was time evolved for a single super-step. The presence of a negative region shows that neither RKL nor RKC is capable of properly handling Dirichlet boundary conditions.

To illustrate the insufficiency of other STS schemes, we use second-order Runge-Kutta-Legendre (RKL2) [1] and second-order Runge-Kutta-Chebyshev (RKC2) [4] schemes to solve the 1D heat equation for a copper rod with thermal diffusivity $\alpha = 1.166$ cm^2/s. In Figure 1, we illustrate the resulting numerical solutions from RKL2 and

RKC2 after starting with a delta function initial condition and evolving the system for a single super-step. We see that after a single super-step, the numerical schemes RKL2 and RKC2 clearly fail to be monotone by yielding negative temperatures. This strongly indicates the need for a new scheme which can handle the boundary condition, which is what we turn our attention to in the next section.

2. Monotone Stability Analysis

To compare our method to other STS schemes and demonstrate its necessity, we will consider a 2-stage STS scheme for a heat equation problem with Dirichlet boundary conditions. To impose this boundary condition, assume that the solution value at the left endpoint u_0^n is fixed at some value, thus $u_0^n = u(x_0, t_n) = u(x_0, t_0)$. For a general 2-stage STS scheme, we can conceptualize the scheme as two explicit Forward Euler (FE) time steps of different sizes. Denoting the time steps as Δt_1 and Δt_2 of the first and second stages respectively and writing the heat equation as

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}, \tag{5}$$

FE gives us the following two equations for each stage:

$$\begin{aligned} \frac{u_i^{n+1} - u_i^n}{\Delta t_1} &= \alpha \frac{u_{i-1}^n - 2u_i^n + u_{i+1}^n}{\Delta x^2} \\ \frac{u_i^{n+2} - u_i^{n+1}}{\Delta t_2} &= \alpha \frac{u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1}}{\Delta x^2}. \end{aligned} \tag{6}$$

Using the first of these equations, we can express the point adjacent to the boundary u_1^{n+2} as:

$$u_1^{n+2} = (\delta_1 + \delta_2 - 2\delta_1\delta_2)\, u_0^n + (1 - 2\delta_1 - 2\delta_2 + 5\delta_1\delta_2)\, u_1^n + (\delta_1 + \delta_2 - 4\delta_1\delta_2)\, u_2^n + \delta_1\delta_2 u_3^n, \tag{7}$$

where $\delta_1 = \Delta t_1 \alpha/\Delta x^2$ and $\delta_2 = \Delta t_2 \alpha/\Delta x^2$ are scaled substeps. The values we choose for δ_1 and δ_2 will determine our numerical scheme and its properties.

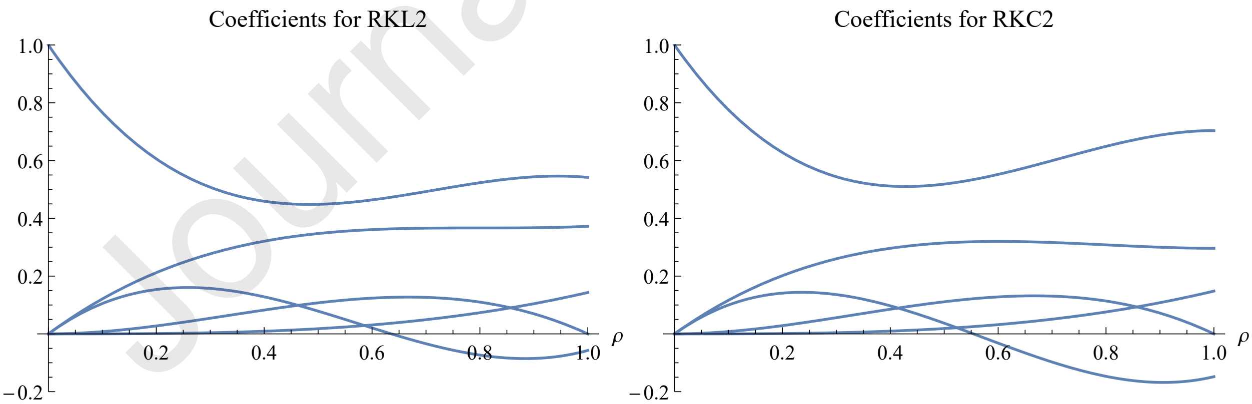


Fig. 2
A plot of the coefficients of the 3 stage RKL2 and 3 stage RKC2 scheme. The negative region indicates a loss of the CMP on both of them.

For instance, we can guarantee our scheme will be monotone by imposing equation (4) as a constraint, which is equivalent to requiring that the coefficients for u_0^n, u_1^n, etc. must all be non-negative. If we plot the values of δ_1, δ_2 that satisfy these constraints, we obtain the orange region given in Fig. 3. Had we performed this derivation in the absence of BC, the values of δ_1, δ_2 for which equation (4) is satisfied would be given by the blue region. We say a STS scheme given by (δ_1', δ_2') has the CMP if the line between $(\delta_1, \delta_2) = (0, 0)$ and $(\delta_1, \delta_2) = (\delta_1', \delta_2')$ is contained within the region of monotone stability. For an s-stage STS scheme, the procedure outlined above can be followed to derive an equation that is analogous to equation (7), in which case the coefficients will be functions of $\delta_1, \delta_2, \ldots, \delta_s$. Throughout the rest of this paper, we will use the term "coefficients" to refer to these functions of δ_1, δ_2, etc. for other STS schemes. Supposing we followed such a procedure for an s-stage scheme, we could denote $c_i(\delta_1, \ldots, \delta_s)$ as the coefficient for u_i^n. A convenient way to visually determine whether an STS scheme is monotone or convex monotone is to plot the values of the coefficients along the line from the origin to the point $\vec{\delta}' = (\delta_1', \ldots, \delta_s')$ specifying the scheme. To do this, we set $(\delta_1, \ldots, \delta_s) = \rho(\delta_1', \ldots, \delta_s')$, where ρ is a parameter we can vary from zero to one, and thus turn each c_i into a univariate function $c_i(\rho) \equiv c_i(\rho\delta_1', \ldots, \rho\delta_s')$. If any of these functions is ever negative for $0 \le \rho \le 1$, then the method does not have the CMP, and if one of the functions is negative for $\rho = 1$, then the method is not monotone either. To illustrate, we plot in Figure 2 the coefficient functions for 3-stage RKL2 and 3-stage RKC2 when Dirichlet boundary conditions are imposed. Due to the negative regions in both plots, we can infer that these methods do not have the CMP and are not even monotone, which explains why these methods gave negative temperatures in Figure 1.

We define the optimal monotone STS scheme for the BC problem as the configuration of δ_1, δ_2 that is within the orange region and maximizes the total time step $\tau = \Delta t_1 + \Delta t_2$. Similarly, the optimal convex monotone STS scheme is the configuration of δ_1, δ_2 that has the CMP and maximizes τ. Because the orange region is subsumed by the blue region, the optimal scheme with the BC (for both monotone and convex monotone) will have a smaller total time step than the optimal scheme without the BC.

Without the BC, we see from the blue region that the optimal monotone scheme is given by the blue asterisk, and the optimal convex monotone scheme is given by the red asterisk. These schemes have been studied before and are known as 2-stage Runge-Kutta-Chebyshev (RKC1) [4] and Runge-Kutta-Legendre (RKL1) [1] respectively. With the BC, the optimal monotone scheme is given by the green asterisk, which corresponds to

$$(\delta_1, \delta_2) = (1/3, 1)\,, \tag{8}$$

and thus the total time step is

$$\tau = \delta_1 \frac{\Delta x^2}{\alpha} + \delta_2 \frac{\Delta x^2}{\alpha} = \frac{2}{3}\Delta t_{exp} + 2\Delta t_{exp} = \frac{8}{3}\Delta t_{exp}, \tag{9}$$

where $\Delta t_{exp} = \Delta x^2/2\alpha$ is the maximal time step allowed in Forward Euler that maintains monotone stability. We will call this scheme 2-stage RKU1. At two stages, RKU1 requires the same number of operations as two steps of FE but integrates $\tau = \frac{8}{3}\Delta t_{exp}$ instead of $\tau = 2\Delta t_{exp}$. The amount that our STS scheme can integrate over FE for the same amount of computational work tells us how much advantage the former has over the latter. For instance, the RKC1 scheme is given by

$$\delta_1, \delta_2 = \frac{2+\sqrt{2}}{2}, \frac{2-\sqrt{2}}{2} \tag{10}$$

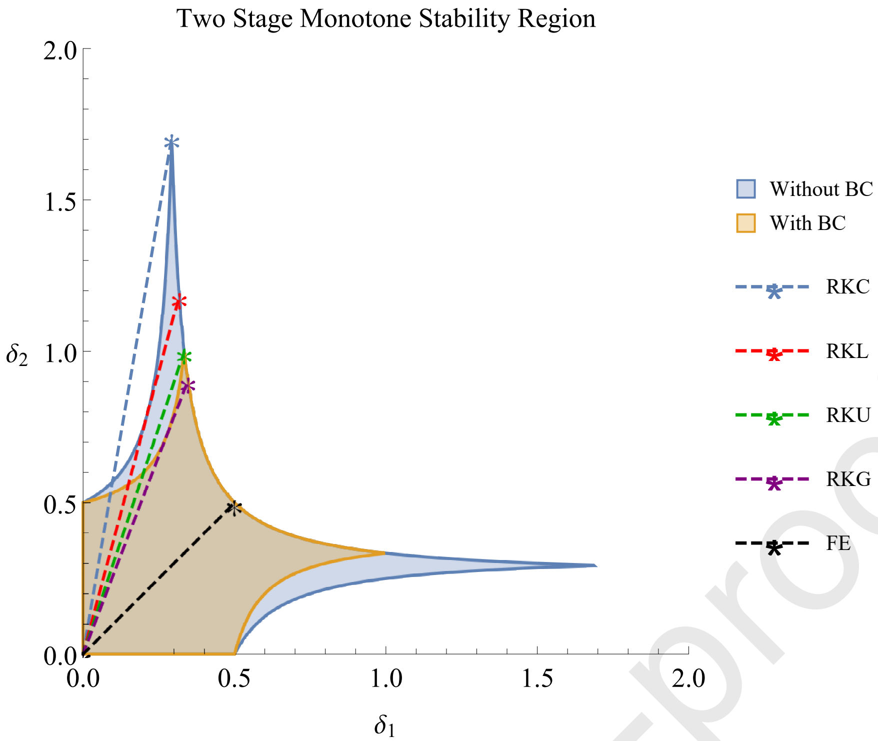


Fig. 3
A plot of the regions in the δ_1-δ_2 plane which give 2-stage STS schemes which are monotone stable with no boundary conditions (blue) and with boundary conditions (orange)

and has total time step $\tau = 4\Delta t_{exp}$. Thus, RKC1 has a greater advantage over RKU1, but RKC1 has the disadvantage that it will not necessarily be monotone once a Dirichlet BC is imposed. For 3-stage RKU1, the time steps are

$$\Delta t_1 = (2 - \sqrt{2})\Delta t_{exp} \tag{11}$$

$$\Delta t_2 = \Delta t_{exp} \tag{12}$$

$$\Delta t_3 = (2 + \sqrt{2})\Delta t_{exp} \tag{13}$$

which gives

$$\tau = 5\Delta t_{exp}. \tag{14}$$

In general, we find that the scaling of τ is quadratic with the number of stages

$$\tau = \frac{s(s+2)}{3}\Delta t_{exp}. \tag{15}$$

The numerical scheme can be derived for a larger number of stages by using shifted Chebyshev polynomials of the second kind. This differs from RKC1 which uses Chebyshev polynomials of the first kind. Chebyshev polynomials of the second kind are denoted U_n for the polynomial of degree n, hence the name RKU1.

The STS schemes that we will describe are intended to solve the ODE system resulting from the discretization of a parabolic PDE. Let

$$\frac{du}{dt} = \mathbf{M}u(t) \tag{16}$$

be such a ODE system where $\mathbf{M}$ is a constant coefficient matrix that represents the discretization of the PDEs parabolic operator. As an ODE system, the analytic solution to (16) is

$$u(t) = e^{t\mathbf{M}} u(0) = \sum_{n=0}^{\infty} \frac{(t\mathbf{M})^n}{n!} u(0) \tag{17}$$

We can express our numerical scheme by its stability polynomial $R_s(z)$, which acts as an approximation to (17). For an s-stage STS scheme, the stability polynomial will have order s and evolves our numerical solution from time t to $t + \tau$ with the relation

$$u(t + \tau) \approx R_s(\tau\mathbf{M})u(t). \tag{18}$$

A scheme specified by polynomial $R_s(z)$ is p^{th} order accurate when $R_s(z)$ has leading order terms that match the series expansion of e^z up to and including the term of degree p. Throughout this paper, we will use the notation conventions in [1, 7] to specify the stability polynomial and the other parameters needed to implement a super-time-stepping scheme. The stability polynomial for an RKU1 scheme with s stages is given by

$$R_s(z) = b_s U_s (1 + w_1 z) \tag{19}$$

$$b_s = \frac{1}{(s+1)} \tag{20}$$

$$w_1 = \frac{3}{s(s+2)} \tag{21}$$

$$\mu_j = \frac{2b_j}{b_{j-1}}; \quad \nu_j = -\frac{b_j}{b_{j-2}}$$

$$\tilde{\mu}_j = \mu_j w_1 = \mu_j \frac{3}{s(s+2)}.$$

However, as shown in Figure 4, this scheme does not satisfy the CMP even at 2 stages. As such, we now turn our

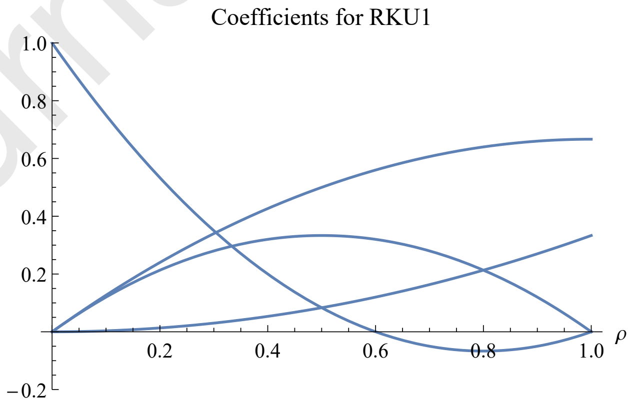


Fig. 4
A plot of the coefficients of the RKU1 scheme, where the negative regions indicate a lack of the CMP.

attention to the scheme which does satisfy the CMP: the Runge-Kutta-Gegenbauer method.

3. RKG1: Runge-Kutta-Gegenbauer method at first-order

With careful analysis of the conditions which lead to the CMP for the Dirichlet Boundary conditions, one finds that this lends rise to a Runge-Kutta-Gegenbauer (RKG) scheme whose stability polynomials are the shifted Gegenbauer Polynomials with $\alpha = \frac{3}{2}$, a particular instantiation of the work described in [5]. The stability polynomial of an s-stage scheme is given by:

$$R_s(z) = a_s + b_s C_s^{(3/2)}(1 + w_1 z) \tag{22}$$

Where $C_s^{3/2}$ is the s-order Gegenbauer polynomial with $\alpha = 3/2$. Imposing the condition that $R_s(0) = 1$ and $R_s'(0) = 1$, it can be easily verified that:

$$b_s = \frac{2}{(s+1)(s+2)} \tag{23}$$

$$w_1 = \frac{4}{s(s+3)}. \tag{24}$$

With $a_s = 0\ \forall s$. At this point it becomes necessary to note that the Gegenbauer Polynomials have a recurrence relation as follows, independent of α:

$$C_s^{\alpha}(z) = \frac{1}{s}\left[2z(s+\alpha-1)C_{s-1}^{\alpha}(z) - (s+2\alpha-2)C_{s-2}^{\alpha}(z)\right]. \tag{25}$$

As this recurrence relation is what allows us to build the recursion scheme for a numerical method. We find that the stability polynomials for the RKG method obey the following:

$$b_j C_j^{(3/2)}(1+w_1 z) = \mu_j b_{j-1} C_{j-1}^{(3/2)}(1+w_1 z) + \nu_j b_{j-2} C_{j-2}^{(3/2)}(1+w_1 z) + \tilde{\mu}_j b_{j-1} C_{j-1}^{(3/2)}(1+w_1 z). \tag{26}$$

Where the parameters are defined as:

$$\mu_j = \frac{2j+1}{j}\frac{b_j}{b_{j-1}}$$
$$\tilde{\mu}_j = \mu_j w_1$$
$$\nu_j = -\frac{(j+1)}{j}\frac{b_j}{b_{j-2}}.$$

Which leads to the following numerical scheme:

$$\begin{aligned}
Y_0 &= u(t_0)\\
Y_1 &= Y_0 + \tilde{\mu}_1 \tau \mathbf{M} Y_0\\
Y_j &= \mu_j Y_{j-1} + \nu_j Y_{j-2} + \tilde{\mu}_j \tau \mathbf{M} Y_{j-1}, 2 \le j \le s\\
u(t_0 + \tau) &= Y_s.
\end{aligned}$$

Where $\mathbf{M}$ is an operator with real, negative eigenvalues.

For a two stage scheme, we find that the values of δ_1, δ_2 that are optimal and satisfy the CMP are:

$$\delta_1 = \frac{5-\sqrt{5}}{8}$$
$$\delta_2 = \frac{5+\sqrt{5}}{8}.$$

4. RKG2: Runge-Kutta-Gegenbauer at second-order

As is done in section 2.3 of [1], second-order accuracy requires that we impose the additional constraint on the stability polynomial, $R_s''(0) = 1$. From there, we find the following constraints:

$$w_1 = \frac{6}{(s+4)(s-1)} \tag{27}$$

$$b_j = \frac{4(j-1)(j+4)}{3j(j+1)(j+2)(j+3)} \tag{28}$$

$$a_j = 1 - \frac{(j+1)(j+2)}{2} b_j. \tag{29}$$

Where we have chosen $b_0 = 1$ and $b_1 = \frac{1}{3}$. Using the recurrence relation found in equation 25, we find that the stability polynomials for the second order RKG method obey the following (after simplifying):

$$\begin{aligned} a_j + b_j C_j^{3/2}(1 + w_1 z) =& \mu_j \left(a_{j-1} + b_{j-1} C_{j-1}^{3/2}(1 + w_1 z)\right) + \nu_j \left(a_{j-2} + b_{j-2} C_{j-2}^{3/2}(1 + w_1 z)\right) \\ &+ \tilde{\mu}_j \left(a_{j-1} + b_{j-1} C_{j-1}^{3/2}(1 + w_1 z)\right) + \left(1 - \mu_j - \nu_j\right) + \tilde{\gamma}_j. \end{aligned}$$

Where we have the following definitions:

$$\begin{aligned} \mu_j &= \frac{2j+1}{j} \frac{b_j}{b_{j-1}} \\ \tilde{\mu}_j &= \mu_j w_1 \\ \nu_j &= -\frac{(j+1)}{j} \frac{b_j}{b_{j-2}} \\ \tilde{\gamma}_j &= -\tilde{\mu}_j a_{j-1}. \end{aligned}$$

This recurrence easily lends itself to the following second order accurate scheme for numerical implementation:

$$\begin{aligned} Y_0 &= u(t_0) \\ Y_1 &= Y_0 + \tilde{\mu}_1 \tau \mathbf{M} Y_0 \\ Y_j &= \mu_j Y_{j-1} + \nu_j Y_{j-2} + \left(1 - \mu_j - \nu_j\right) Y_0 + \tilde{\mu}_j \tau \mathbf{M} Y_{j-1} + \tilde{\gamma}_j \tau \mathbf{M} Y_0, 2 \le j \le s \\ u(t_0 + \tau) &= Y_s. \end{aligned}$$

In general, for a first-order accurate scheme

$$w_1 = \frac{1 + 2\alpha}{s(s + 2\alpha)},$$

and for a second-order accurate scheme

$$w_1 = \frac{3 + 2\alpha}{(s + 2\alpha + 1)(s - 1)}.$$

One then finds that in the limit as $\alpha \to \infty$ that $w_1 \to 1$, and the timestep τ approaches the maximal time step allowed under Forward Euler, Δt_{exp}. This scheme satisfies the CMP, as we have have verified for up to 12 stages with a Mathematica script, and we provide a direct example of RKL, and RKG with 7 stages including the boundary condition in Figure 5.

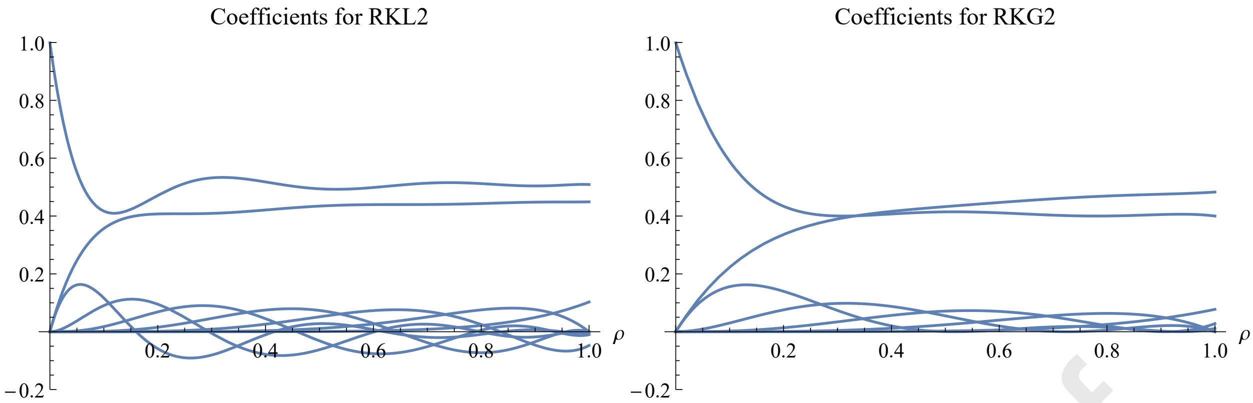


Fig. 5
A plot of the coefficients of 7-stage RKL2 (left) and RKG2 (right) with Dirichlet Boundary Conditions. RKG2 still has the CMP with boundary conditions while RKL2 does not.

5. Convergence Analysis

5.1. One-dimensional heat conduction

In this subsection, we will test that RKG2 indeed converges to the correct solution at second-order by applying this method to one-dimensional heat conduction problems that have analytic solutions. In the first example, we solve the one-dimensional heat equation for two bars of equal length that are brought into contact with each other. Each of the bars is 10 cm in length. Both of the bars are composed of the same material with a spatially-independent thermal diffusivity given by $\alpha = 1\ \mathrm{cm}^2/\mathrm{s}$. Let the temperature of the bars be given by $u(x,t)$. We assume that at $t = 0$ the bar on the left has temperature $u(x,0) = 0\,^\circ$C and the bar on the right has temperature $u(x,0) = 100\,^\circ$C. For our boundary conditions, we assume the left boundary is fixed to $u(-10,t) = 0\,^\circ$C and the right boundary is fixed to $u(10,t) = 100\,^\circ$C. Our initial value problem is therefore

$$u(x,0) = \begin{cases} 0\,^\circ\mathrm{C} & x < 0 \\ 100\,^\circ\mathrm{C} & x \geq 0 \end{cases}, \quad u(-10,t) = 0\,^\circ\mathrm{C}, \quad u(10,t) = 100\,^\circ\mathrm{C} \tag{30}$$

where the boundary conditions hold for all $t > 0$. The exact solution $u(x,t)$ for the temperature field in the domain $-10 < x < 10$ is

$$u(x,t) = \sum_{k=-\infty}^{\infty} \hat{u}(x - 2kL, t) + kT_0 \tag{31}$$

$$\hat{u}(x,t) = \frac{T_0}{2}\left(1 + \mathrm{erf}\left(\tfrac{x}{2\sqrt{\alpha t}}\right)\right) \tag{32}$$

where $L = 10$ cm and $T_0 = 100\,^\circ$C. In Table 1, we have recorded the L^1, L^2, and L^∞ error for RKG2 with varying spatial and time steps showing clear 2nd order convergence for RKG2.

6. Examples and Applications

In this section we present numerical calculations of the heat equation in two dimensions. In general the heat equation reads as:

$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u.$$

Table 1
Errors for the RKG2 method when the 1D heat conduction problem is solved with an s-stage scheme, N_x spatial grid points, and the numerical solution is iterated N_{sts} super-time steps.

N_x	N_{sts}	s	L^1	L^2	L^∞
RKG2					
80	15	3	4.15×10^{-1}	1.84×10^{-1}	1.17×10^{-1}
160	30	5	9.92×10^{-2}	4.35×10^{-2}	2.77×10^{-2}
320	60	7	2.44×10^{-2}	1.07×10^{-2}	6.78×10^{-3}
640	120	10	6.06×10^{-3}	2.65×10^{-3}	1.68×10^{-3}
1280	240	14	1.51×10^{-3}	6.60×10^{-4}	4.18×10^{-4}
2560	480	20	3.77×10^{-4}	1.65×10^{-4}	1.04×10^{-4}

For two dimensions, we use the following centered finite difference discretization:

$$\frac{u^{n+1}(x_i, y_i) - u^n(x_i, y_i)}{\Delta t} = \alpha \frac{u^n(x_{i-1}, y_i) + u^n(x_{i+1}, y_i) + u^n(x_i, y_{i-1}) + u^n(x_i, y_{i+1}) - 4u^n(x_i, y_i)}{\Delta x \Delta y}.$$

In the two dimensional Forward Euler scheme, we find that the maximal time step allowed that is still stable is:

$$\Delta t_{exp} = \frac{\Delta x^2 + \Delta y^2}{8\alpha} = \frac{\Delta x^2}{4\alpha},$$

where the last equality holds when $\Delta x = \Delta y$. In Figure 6, we compare the numerical solutions of 3-stage RKL2 and RKG2 after a single super-step. We start with the initial conditions of $u(x, y, t) = 100\,^\circ$C at the point $(x, y, t) = (0.02$ cm$, 0.97$ cm$, 0)$ and $0\,^\circ$C elsewhere and with boundary conditions $u(x, y, t) = 0\,^\circ$C at $x = 0, 1$ and $u(x, y, t) = 0$ at $y = 0, 1$ for all $t \geq 0$. We use step sizes $\Delta x = \Delta y = .01$ cm. In this problem, RKL2 fails to maintain monotonicity but RKG2 succeeds. This is demonstrated by the fact that the RKL2 solution has regions of negative temperature, which violates equation (3). The negative components regions in RKL2 solution are a direct result of the negative

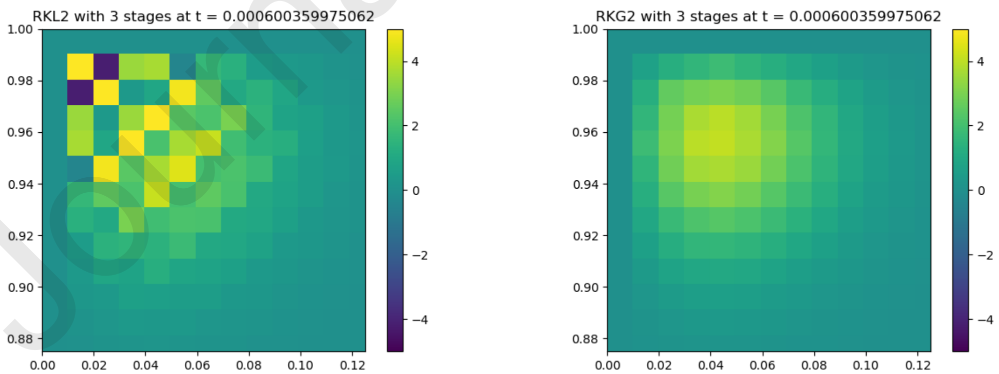


Fig. 6
These plots show the numerical solutions given by RKL2 and RKG2 on the left and right respectively after a single super-step.

region present in Figure 2. It is worth noting that in the asymptotic regime, RKL is linearly stable and will thus approach the correct solution. However at early times, the lack of monotonicity causes a violation of the Second

Law of Thermodynamics - there are areas getting colder and different areas getting hotter without any external energy input. As such, RKG provides a much more reliable method for calculations at early times with Dirichlet boundary conditions.

7. Conclusions

We have shown that current STS methods are incapable of handling boundary conditions and maintaining monotonicity, thus presenting a clear need for a new method. The $\mathrm{RKG}^{3/2}$ method fills this need, maintaining monotonicity throughout calculations while still being capable of performing super-time steps that scale in proportion to the number of stages squared and thus providing a clear advantage over the Forward Euler scheme. Furthermore, this scheme can be used with Strang Splitting, thus allowing for equations which are not purely parabolic PDEs to take advantage of STS on the parabolic aspect of the equation. It is worth mentioning that in particular, RKG allows for accurate modeling of solutions at early times and as such if early times are under investigation RKG is the optimal scheme. However, RKC, RKL, RKU, and RKG are all linearly stable and as such will all approach the correct solution asymptotically at long times. The usefulness of RKL for finding the asymptotic solution should not be ignored in favor of RKG, as RKL will be more computationally efficient even when including boundary conditions.

Acknowledgments

This study was performed under the auspices of the US Department of Energy. Los Alamos National Laboratory, is managed by Triad National Security, LLC for the U.S. Department of Energy, National Nuclear Security Administration, under contract 89233218CNA000001.

References

[1] C. D. Meyer, D. S. Balsara, T. D. Aslam, A stabilized Runge–Kutta–Legendre method for explicit super-time-stepping of parabolic and mixed equations, Journal of Computational Physics 257 (2014) 594–626.
[2] C. D. Meyer, D. S. Balsara, T. D. Aslam, A second-order accurate Super TimeStepping formulation for anisotropic thermal conduction, Monthly Notices of the Royal Astronomical Society 422 (2012) 2101–2115.
[3] V. Alexiades, G. Amiez, P.-A. Gremaud, Super-time-stepping acceleration of explicit schemes for parabolic problems, Communications in numerical methods in engineering 12 (1996) 31–42.
[4] J. G. Verwer, Explicit Runge-Kutta methods for parabolic partial differential equations, Applied Numerical Mathematics 22 (1996) 359–379.
[5] S. O'Sullivan, Runge-Kutta-Gegenbauer Explicit Methods for Advection-Diffusion Problems, Journal of Computational Physics 388 (2019) 209–223.
[6] R. J. LeVeque, Numerical methods for conservation laws, volume 3, Springer, 1992.
[7] J. Verwer, W. Hundsdorfer, B. Sommeijer, Convergence properties of the runge-kutta-chebyshev method, Numerische Mathematik 57 (1990) 157–178.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Conflicts of interests

The authors declare that they have no known conflicts of interest related to the work reported in this paper.

Conflicts of interests

The authors declare that they have no known conflicts of interest related to the work reported in this paper.

Tariq Aslam: Conceptualization, Writing – Review and Editing, Visualization, Supervision, Project administration, Funding acquisition

Chad Meyer: Conceptualization, Writing – Review and Editing, Visualization, Supervision

Torrey Saxton: Software, Writing – Original Draft, Visualization

Tim Skaras: Software, Writing – Original Draft, Writing – Review and Editing, Visualization