On Global Error Estimation and Control of Finite Difference Solutions for Parabolic Equations

KRISTIAN DEBRABANT AND JENS LANG

Abstract. The aim of this paper is to extend the global error estimation and control addressed in Lang and Verwer [SIAM J. Sci. Comput. 29, 2007] for initial value problems to finite difference solutions of parabolic partial differential equations. The classical ODE approach based on the first variational equation is combined with an estimation of the PDE spatial truncation error to estimate the overall error in the computed solution. Control in a discrete $L^2$-norm is achieved through tolerance proportionality and mesh refinement. Numerical examples are used to illustrate the reliability of the estimation and control strategies.

1. Introduction

We consider initial boundary value problems of parabolic type, which can be written as

$$
\partial_t u(t, x) = f(t, x, u(t, x), \partial_x u(t, x), \partial_{xx} u(t, x)) , \quad t \in (0, T], \ x \in \Omega \subset \mathbb{R}^d ,
$$
equipped with an appropriate system of boundary conditions and with the initial condition

$$
u(0, x) = u_0(x) , \quad x \in \overline{\Omega} .
$$
The PDE is assumed to be well posed and to have a unique continuous solution $u(t, x)$ which has sufficient regularity.

The method of lines is used to solve (1) numerically. We first discretize the PDE in space by means of finite differences on a (possibly non-uniform) spatial mesh $\Omega_h$ and solve the resulting system of ODEs using existing time integrators. For simplicity, we shall assume that this system of time-dependent ODEs can be written in the general form

$$
U'_h(t) = F_h(t, U_h(t)) , \quad t \in (0, T] ,
$$
with a unique solution vector $U_h(t)$ being a grid function on $\Omega_h$. Let

$$
R_h : u(t, \cdot) \rightarrow R_h u(t)
$$
be the usual restriction operator defined by $R_h u(t) = (u(t, x_1), \ldots, u(t, x_N))^T$, where $x_i \in \Omega_h$ and $N$ is the number of all mesh points. Then we take as initial condition $U_{h,0} = R_h u(0)$.

To solve the initial value problem (3), we apply a numerical integration method at a certain time grid

$$
0 = t_0 < t_1 < \cdots < t_n < \cdots < t_{M-1} < t_M = T ,
$$

Mathematics Subject Classification. 65M15, 65M20, 65M60.

Key words and phrases. numerical integration for PDEs; finite difference method; global error estimation; global error control; defects and local errors; tolerance proportionality.
using local control of accuracy. This yields approximations $V_h(t_n)$ to $U_h(t_n)$, which may be calculated for other values of $t$ by using a suitable interpolation method provided by the integrator. The global time error is then defined by

\begin{equation}
E_h(t) = V_h(t) - U_h(t) .
\end{equation}

Numerical experiments in [4] for ODE systems have shown that classical global error estimation based on the first variational equation is remarkably reliable. In addition, having the property of tolerance proportionality, that is, there exists a linear relationship between the global time error and the local accuracy tolerance, $e_h(t)$ can be successfully controlled by a second run with an adjusted local tolerance. Numerous techniques to estimate global errors are described in [8].

In order for the method of lines to be used efficiently, it is necessary to take also into account the spatial discretization error. Defining the spatial discretization error by

\begin{equation}
\eta_h(t) = U_h(t) - R_h u(t) ,
\end{equation}

the vector of overall global errors $E_h(t) = V_h(t) - R_h u(t)$ may be written as sum of the global time and spatial error, that is,

\begin{equation}
E_h(t) = e_h(t) + \eta_h(t) .
\end{equation}

It is the purpose of this paper to present a new error control strategy for the global errors $E_h(t)$. We will mainly focus on reliability. So our aim is to provide error estimates $\tilde{E}_h(t) \approx E_h(t)$ which are not only asymptotically exact, but also work reliably for moderate tolerances, that is for relatively coarse discretizations.

The global errors are measured in discrete $L^2$-norms. A priori bounds for the global error in such norms are well known, see e.g. [5, 9]. However, reliable a posteriori error estimation and efficient control of the accuracy of the solution numerically computed to an imposed tolerance level are still challenging. We achieve global error control by iteratively improving the temporal and spatial discretizations according to estimates of $e_h(t)$ and $\eta_h(t)$. The global time error is estimated and controlled along the way fully described in [4]. To estimate the global spatial error, we follow an approach proposed in [1] (see also [6]) and use Richardson extrapolation to set up a linearised error transport equation.

2. Spatial and time error

By making use of the restriction operator $R_h$, the spatial truncation error is defined by

\begin{equation}
\alpha_h(t) = (R_h u)'(t) - F_h(t, R_h u(t)) .
\end{equation}

From (3) and (9), it follows that the global spatial error $\eta_h(t)$ representing the accumulation of the spatial discretization error is the solution of the initial value problem

\begin{equation}
\eta_h(t) = F_h(t, U_h(t)) - F_h(t, R_h u(t)) - \alpha_h(t) , \quad t \in (0, T] ,
\eta_h(0) = 0 .
\end{equation}

Assuming $F_h$ to be continuously differentiable, the mean value theorem for vector functions yields

\begin{equation}
\eta_h(t) = \partial U_h F_h(t, U_h(t)) \eta_h(t) - \alpha_h(t) + O(\eta_h(t)^2) , \quad t \in (0, T] ,
\eta_h(0) = 0 .
\end{equation}
With \( V_h(t) \) being the continuous extension of the numerical approximation to \( [3] \), the residual time error is defined by
\[
\text{(12)} \quad r_h(t) = V_h'(t) - F_h(t, V_h(t)).
\]
Thus the global time error \( e_h(t) \) fulfills the initial value problem
\[
\text{(13)} \quad \begin{align*}
e_h'(t) &= F_h(t, V_h(t)) - F_h(t, U_h(t)) + r_h(t), & t \in (0, T], \\
e_h(0) &= 0.
\end{align*}
\]
Again, the mean value theorem yields
\[
\text{(14)} \quad \begin{align*}
e_h'(t) &= \partial U_h F_h(t, V_h(t)) e_h(t) + r_h(t) + O(e_h(t)^2), & t \in (0, T], \\
e_h(0) &= 0.
\end{align*}
\]
 Apparently, by implementing proper choices of the defects \( \alpha_h(t) \) and \( r_h(t) \), solving \( [11] \) and \( [14] \) will in leading order provide approximations to the true global error. The issue of how to approximate the spatial truncation error and the residual time error will be discussed in the next sections.

3. Estimation of the residual time error

We assume that the time integration method used to approximate the general ODE system \( [3] \) is of order \( p \leq 3 \). Following the approach proposed in \( [3] \) we define the interpolated solution \( V_h(t) \) by piecewise cubic Hermite interpolation. Let \( V_{h,n} = V_h(t_n) \) and \( F_{h,n} = F_h(t_n, V_{h,n}) \) for all \( n = 0, 1, \ldots, M \). Then at every subinterval \([t_n, t_{n+1}]\) we form
\[
\text{(15)} \quad V_h(t) = V_{h,n} + A_n(t - t_n) + B_n(t - t_n)^2 + C_n(t - t_n)^3, \quad t_n \leq t \leq t_{n+1},
\]
and choose the coefficients such that \( V_h'(t_n) = F_{h,n} \) and \( V_h'(t_{n+1}) = F_{h,n+1} \). This gives
\[
\text{(16)} \quad V_h(t_n + \theta \tau_n) = v_0(\theta)V_{h,n} + v_1(\theta)V_{h,n+1} + \tau_n w_0(\theta)F_{h,n} + \tau_n w_1(\theta)F_{h,n+1}
\]
with \( 0 \leq \theta \leq 1, \tau_n = t_{n+1} - t_n \), and
\[
\text{(17)} \quad v_0(\theta) = (1 - \theta)^2(1 + 2\theta), \quad v_1(\theta) = \theta^2(3 - 2\theta), \quad w_0(\theta) = (1 - \theta)^2 \theta, \quad w_1(\theta) = \theta^2(\theta - 1).
\]
Now let \( Y_h(t) \) be the (sufficiently smooth) solution of the ODE \( [3] \) with initial value \( Y(t_n) = V_{h,n} \). Then the local error of the time integration method at time \( t_{n+1} \) is given by
\[
\text{(18)} \quad l e_{n+1} = V_{h,n+1} - Y_h(t_{n+1}) = O(\tau_n^{p+1}).
\]
Combining \( [16] \) and \( [18] \) gives
\[
\text{(19)} \quad V_h(t_n + \theta \tau_n) - Y_h(t_n + \theta \tau_n) = v_1(\theta) l e_{n+1} - Y_h(t_n + \theta \tau_n) + v_0(\theta)Y_h(t_n) + v_1(\theta)Y_h(t_{n+1})
\]
\[
+ \tau_n w_0(\theta)F_h(t_n, V_h(t_n)) + \tau_n w_1(\theta)F_h(t_{n+1}, Y_h(t_{n+1})),
\]
and by Taylor expansion we obtain
\[
\text{(20)} \quad V_h(t_n + \theta \tau_n) - Y_h(t_n + \theta \tau_n) = v_1(\theta) l e_{n+1} + \frac{1}{24}(2\theta^3 - 3\theta^2 + \theta) \tau_n^4 Y_h^{(4)}(t_n) + O(\tau_n^{p+2}).
\]
Recalling \( Y_h'(t) = F_h(t, Y_h(t)) \) for \( t \in (t_n, t_{n+1}] \) and rewriting the residual time error as
\[
\text{(21)} \quad r_h(t) = V_h'(t_n + \theta \tau_n) - Y_h'(t_n + \theta \tau_n) + F_h(t, Y_h(t)) - F_h(t, V_h(t)),
\]
with \( \theta = (t - t_n)/\tau_n \), we find by differentiating the right hand side of (20)

\[
(22) \quad r_h(t_n + \theta \tau_n) = 6(\theta - \theta^2) \frac{\ell e_{n+1}}{\tau_n} + \frac{1}{12} (3\theta^2 - \theta - 2\theta^3) \tau_n^3 Y_h^{(4)}(t_n) + O(\tau_n^{p+1}).
\]

Setting \( \theta = 1/2 \) in (22) will reveal

\[
(23) \quad r_h(t_{n+1/2}) = \frac{3}{2} \frac{\ell e_{n+1}}{\tau_n} + O(\tau_n^{p+1}).
\]

Thus the cubic Hermite defect halfway the step interval can be used to retrieve in leading order the local error of any one-step method of order \( 1 \leq p \leq 3 \) (see also [4], Section 2.2). Following the arguments given in [4], Section 2.1, we consider instead of (14) the step size frozen version

\[
(24) \quad \tilde{e}_h(t) = \partial_{U_h} F_h(t, U_{h,n}) \tilde{e}_h(t) + \frac{2}{3} r_h(t_{n+1/2}), \quad t \in (t_n, t_{n+1}], \quad n = 0, \ldots, M-1,
\]

\[
\tilde{e}_h(0) = 0
\]

to approximate the global time error \( e_h(t) \). Using

\[
(25) \quad V_h(t_{n+1/2}) = \frac{1}{2} (V_{h,n} + V_{h,n+1}) + \frac{\tau}{8} (F_{h,n} - F_{h,n+1})
\]

and

\[
(26) \quad V_h'(t_{n+1/2}) = \frac{3}{2\tau} (V_{h,n+1} - V_{h,n}) - \frac{1}{4} (F_{h,n} + F_{h,n+1})
\]

we can compute the residual time error halfway the step interval from (12)

\[
(27) \quad r_h(t_{n+1/2}) = \frac{3}{2\tau} (V_{h,n+1} - V_{h,n}) - \frac{1}{4} (F_{h,n} + F_{h,n+1}) - F_h \left( t_{n+1/2}, \frac{1}{2} (V_{h,n} + V_{h,n+1}) + \frac{\tau}{8} (F_{h,n} - F_{h,n+1}) \right).
\]

**Remark 3.1.** From (22) we deduce

\[
(28) \quad \frac{1}{\tau_n} \int_{t_n}^{t_{n+1}} r_h(t) \, dt = \frac{\ell e_{n+1}}{\tau_n} + O(\tau_n^{p+1}),
\]

showing, in the light of (23), that \( \frac{2}{3} r_h(t_{n+1/2}) \) is in leading order equal to the time-averaged residual. Thus, we can justify the use of the error equation (24) without the link to the first variational equation. \( \diamond \)

### 4. Estimation of the Spatial Truncation Error

An efficient strategy to estimate the spatial truncation error by Richardson extrapolation is proposed in [1]. We will adopt this approach to our setting.

Suppose we are given a second semi-discretization of the PDE system (1), now with doubled local mesh sizes \( 2h \),

\[
U'_{2h}(t) = F_{2h}(t, U_{2h}(t)), \quad t \in (0, T],
\]

\[
U_{2h}(0) = U_{2h,0}.
\]

In practice, one first chooses \( \Omega_{2h} \) and constructs then \( \Omega_h \) through uniform refinement. The following two assumptions will be needed. (i) The solution \( U_{2h}(t) \) to the discretized PDE on the coarse mesh \( \Omega_{2h} \) exists and is unique. (ii) The spatial discretization error \( \eta_h(t) \) is of
Assuming the term on the right-hand side involving the global time error to be sufficiently small, we obtain by using again Taylor expansion it follows that

\[ U_h^e(t) = U_h(t) - \frac{1}{2^q} \eta_2h(t) + \mathcal{O}(h^{q+1}). \]

The spatial truncation error on the coarse mesh \( \Omega_{2h} \) is analogously defined to (9) as

\[ \alpha_{2h}(t) = (R_{2h} u')(t) - F_{2h}(t, R_{2h} u(t)). \]

Substituting \( R_{2h} u(t) \) from (32) into the right-hand side, using the ODE system (29) to replace \( U_{2h}(t) \), and manipulating the expressions with (33) we get

\[
\alpha_{2h}(t) = \frac{2^q}{2^q - 1} \left( (U_h^c)'(t) - F_{2h}(t, U_h^c(t)) + \mathcal{O}(h^{q+1}) \right) + \frac{1}{2^q - 1} \left( F_{2h}(t, U_{2h}(t) - \eta_2h(t)) + \mathcal{O}(h^{q+1}) \right) - F_{2h}(t, U_{2h}(t)) + \mathcal{O}(h^{q+1}).
\]

Taylor expansions yield

\[ \alpha_{2h}(t) = \frac{2^q}{2^q - 1} \left( (U_h^c)'(t) - F_{2h}(t, U_h^c(t)) \right) + \mathcal{O}(h^{q+1}). \]

Analogously to (8), we set \( e_h^c(t) = V_h^c(t) - U_h^c(t) \). Substituting \( (U_h^c)'(t) \) by \( R_{2h} h(t, U_h^c(t)) \) and using again Taylor expansion it follows that

\[
\alpha_{2h}(t) = \frac{2^q}{2^q - 1} \left( R_{2h} h(t, V_h(t)) - F_{2h}(t, V_h^c(t)) \right) + \mathcal{O}(h^{q+1})
\]

and

\[
\alpha_{2h}(t) = \frac{2^q}{2^q - 1} \left( R_{2h} h(t, V_h(t)) - F_{2h}(t, V_h^c(t)) \right) + \mathcal{O}(e_h(t)^2).
\]

Assuming the term on the right-hand side involving the global time error to be sufficiently small, we can use

\[ \tilde{\alpha}_{2h}(t) = \frac{2^q}{2^q - 1} \left( R_{2h} h(t, V_h(t)) - F_{2h}(t, V_h^c(t)) \right) \]

as approximation for the spatial truncation error on the coarse mesh. To guarantee a suitable quality of the estimation (33) we shall first control the global time error for attempting that afterwards the overall error is dominated by the spatial truncation error (see Section 5).

An approximation \( \tilde{\phi}_{2h}(t) \) of the spatial truncation error on the (original) fine mesh is obtained by interpolation respecting the order of accuracy (see Section 5). Thus, to approximate the global spatial error \( \eta_h(t) \) we consider instead of (11) the step-size frozen version

\[
\eta_h^c(t) = \partial_{U_h} F_h(t_n, V_{h,n}) \eta_h(t) - \tilde{\alpha}_h(t), \quad t \in (t_n, t_{n+1}], \ n = 0, \ldots, M-1,
\]

and

\[ \eta_h(0) = 0. \]
Remark 4.1. If an approximation $\tilde{e}_h(t)$ of the global time error has already been computed, we could make use of $U_h^e(t) \approx V_h^e(t) + \tilde{e}_h^e(t)$ to obtain a better approximation of $\sigma_2h(t)$ from (36). However, we have found by experiments that even in the case when the global time error was not small, using the step size frozen equations (24) and (39) together with (36) to approximate the global time and spatial error does not yield a significantly better approximation. Since in practice the use of formula (36) requires additional function evaluations, equation (38) appears to be more efficient.

Remark 4.2. We note that special care has to be taken in the handling with the spatial truncation error at the boundary when derivative boundary conditions are present. This requests interpolation adopted to the correct order of accuracy, see [1].

5. The example discretization formulas

In order to keep the illustration as simple as possible we restrict ourselves to one space dimension. For the spatial discretization of (11) we use standard second-order finite differences. Hence we have $q=2$. The discrete $L_2$-norm on a non-uniform mesh

$$\|y\|^2 = \sum_{i=1}^{N} \frac{h_i + h_{i+1}}{2} y_i^2,$$

where $y_0$ and $y_{N+1}$ which are given by the boundary values are not considered.

The example time integration formulas are taken from [4]. For the sake of completeness we shall give a short summary of the implementation used. To generate the time grid (5) we use the implementations

$$\tau_0 = \min \left(1.5, \max \left(\frac{2}{3}, 0.9 \frac{r}{\gamma} \right) \tau_n \right), \quad r = \left(\frac{\text{Tol}_n}{D_n}\right)^{1/3}.$$

After each step size change we adjust $\tau_{new}$ to $\tau_{n+1} = (T - t_n)/(1 + (T - t_n)/\tau_{new})$ so as to guarantee to reach the end point $T$ with a step of averaged normal length. The initial step size $\tau_0$ is prescribed and is adjusted similarly.

The linear error transport equations (24) and (39) are simultaneously solved by means of the implicit midpoint rule, which gives approximations $\tilde{e}_{h,n}$ and $\tilde{\eta}_{h,n}$ to the global time and spatial error at time $t = t_n$. We use the implementations

$$\left(I_h - \frac{1}{2} \tau_n A_{h,n}\right) \delta e_{n+1} = 2\tilde{e}_{h,n} + \frac{2}{3} \tau_n r(t_{n+1/2}), \quad \tilde{e}_{h,n+1} = \delta e_{n+1} - \tilde{e}_{h,n},$$

for a vector $y = (y_1, \ldots, y_N)^T \in \mathbb{R}^N$ is defined through

$$\|y\|^2 = \sum_{i=1}^{N} \frac{h_i + h_{i+1}}{2} y_i^2.$$
\[(I_h - \frac{1}{2} \tau_n A_{h,n}) \delta \eta_{n+1} = 2 \hat{\eta}_{h,n} - \tau_n \hat{\alpha}_h(t_{n+1/2}), \]

Clearly, the matrices \(A_{h,n}\) already computed within ROS3P can be reused. The spatial truncation error \(\hat{\alpha}_{2h}(t)\) at \(t = t_{n+1/2}\) is given by

\[(46)\quad \hat{\alpha}_{2h}(t_{n+1/2}) = \frac{4}{3} \left( R_{2h}^{h} F_h \left( t_{n+1/2}, V_h(t_{n+1/2}) \right) - F_{2h} \left( t_{n+1/2}, R_{2h}^{h} V_h(t_{n+1/2}) \right) \right).\]

Since \(V_h(t_{n+1/2})\) and \(F_h(t_{n+1/2}, V_h(t_{n+1/2})\) are available from the computation of \(r_h(t_{n+1/2})\) in (47), this requires only one function evaluation on the coarse grid. The vector \(\hat{\alpha}_{2h}(t_{n+1/2})\) on the coarse mesh is prolongated to the fine mesh and is then divided by \(2^q = 4\) if the neighboring fine grid points are equidistant, otherwise it is divided by \(2^{q-1} = 2\). The remaining \(\hat{\alpha}_h(t_{n+1/2})\) on the fine mesh are computed by interpolation respecting the order of the neighboring spatial truncation errors.

Due to freezing the coefficients in each time step, the second-order midpoint rule is a first-order method when interpreted for solving the linearised equations (14) (or likewise the first variational equation) and (11). Thus if all is going well, we asymptotically have \(\hat{\epsilon}_{h,n} = e_h(t_n) + O(\tau_{\text{max}}^4)\) and \(\hat{\eta}_{h,n} = \eta_h(t_n) + O(\tau_{\text{max}} h_{\text{max}}^q) + O(h_{\text{max}}^{q+1})\).

After computing the spatial truncation errors we can solve the discretized transport errors (45) for all \(\hat{\eta}_{h,n}\). We shall distinguish between two different mesh adaptation approaches: (i) globally uniform and (ii) locally adaptive refinement. Although the uniform strategy may be less efficient, it is very easy to implement and therefore of special practical interest if software packages which do not allow dynamic adaptive mesh refinement are used.

**Uniform spatial refinement.** Let \(Tol\) be a given tolerance. Then our aim is to guarantee \(\|\eta_h(T)\| \leq Tol\). From (45), we get an approximate value \(\hat{\eta}_{h,M}\) for the spatial discretization error at \(T\). If the desired accuracy is still not satisfied, i.e., \(\|\hat{\eta}_{h,M}\| > Tol\), we choose a new (uniform) spatial resolution

\[(47)\quad h_{\text{new}} = \sqrt[3]{\frac{Tol}{\|\hat{\eta}_{h,M}\|}} h\]

to account for achieving \(\|\eta_{h_{\text{new}}}(T)\| \approx Tol\). From \(h_{\text{new}}\) we determine a new number of mesh points. The whole computation is redone with the new spatial mesh.

**Adaptive spatial refinement.** The main idea of our local spatial mesh control is based on the observation that the principle of tolerance proportionality can be also applied to the spatial discretization error. Multiplying all \(\hat{\alpha}_h(t_{n+1/2})\) in (45) by a certain constant multiplies all \(\hat{\eta}_{h,n+1}\) by the same constant since \(\hat{\eta}_{h,0} = 0\). Set \(Tol_a^h = Tol_A + Tol_R^\|V_{h,n}\|\) where \(Tol_A^h\) and \(Tol_R^\|V_{h,n}\|\) are given local tolerances and define a local estimator \(A_n\) through

\[(48)\quad A_n^2 = \sum_{i : x_i \in F_h} \|2 h_i |\hat{\alpha}_i(t_{n+1/2})|^2,\]

where \(F_h\) denotes the set of all (fine) mesh points that do not belong to the coarse mesh. Remember we have second order of the spatial truncation error in these points. If \(A_n \leq Tol_a^h\) the mesh is only coarsened. Otherwise, if \(A_n > Tol_a^h\) the mesh is improved by refinement and coarsening as well. We set \(\alpha_{\text{tol}} = 0.9 Tol_a^h / \sqrt{N}\) and mark all \(x_i \in F_h\)

\[(49)\quad \begin{align*}
\text{for refinement if } & \sqrt{h_i |\hat{\alpha}_i(t_{n+1/2})|} > \alpha_{\text{tol}} \\
\text{and for coarsening if } & \sqrt{h_i |\hat{\alpha}_i(t_{n+1/2})|} < 0.1 \alpha_{\text{tol}}.
\end{align*}\]

\[\]
Grid adaptation is first performed for the coarse mesh and afterwards the fine mesh is constructed by halving each interval. If \( x_i \) is marked for refinement the corresponding coarse grid interval is halved. Grid points are only removed if there are two equidistant neighboring intervals the midpoints of which are marked for coarsening. Finally, the grid is smoothed such that \( 0.5 \leq h_i/h_{i-1} \leq 2 \) everywhere. Data transfer from old to new meshes is done by cubic Hermite interpolation where the necessary first derivatives are determined from fourth order finite differences.

After mesh adaptation the local time step is redone with the new mesh. The procedure is continued until first \( D_n \leq Tol_n \) and second \( A_n \leq Tol_n \) hold. The whole strategy aims at equidistributing the local values \( \sqrt{h_i} \alpha_i(t_{n+1/2}) \). Asymptotically we get

\[
A_n \approx \left( 2 \sum_{i : x_i \in F_h} \alpha_{tol}^2 \right)^{1/2} = \left( 2 \sum_{i : x_i \in F_h} \frac{0.81 (Tol_n^a)^2}{N} \right)^{1/2} \approx 0.9 Tol_n^a ,
\]

where the factor 0.9 improves the robustness of the equidistribution principle.

6. The control rules

Like for the ODE case studied in [4] our aim is to provide global error estimates and to control the accuracy of the solution numerically computed to the imposed tolerance level. Let \( GTol_A \) and \( GTol_R \) be the global tolerances. Then we start with the local tolerances \( Tol_A = GTol_A, Tol_R = GTol_R \), and in the spatially adaptive case also with \( Tol_n^A = C_{\alpha} GTol_A \), and \( Tol_n^R = C_{\alpha} GTol_R \), where the factor \( C_{\alpha} > 1 \) ensures that the residual time error is small with respect to the spatial truncation error and therefore the use of (38) is justified.

Suppose the numerical schemes have delivered an approximate solution \( V_{h,M} \) and global estimates \( \tilde{e}_{h,M} \) and \( \tilde{\eta}_{h,M} \) for the time and spatial error at time \( t_M = T \). We then verify whether

\[
\| \tilde{e}_{h,M} \| \leq C_T C_{\text{control}} Tol_M, \quad Tol_M = GTol_A + GTol_R \| V_{h,M} \| ,
\]

where \( C_{\text{control}} \approx 1, \) typically \( > 1, \) and \( C_T \in (0, 1) \) denotes the fraction desired for the global time error with respect to the tolerance \( Tol_M \). If (51) does not hold, the whole computation is redone over \([0, T]\) with the same initial step \( \tau_0 \) and the adjusted local tolerances

\[
Tol_A = Tol_A \cdot fac, \quad Tol_R = Tol_R \cdot fac, \quad fac = C_T Tol_M / \| \tilde{e}_{h,M} \| .
\]

Based on tolerance proportionality, reducing the local error estimates with the factor \( fac \) will decrease \( e_h(T) \) by \( fac \) [7].

If (51) holds, we check whether

\[
\| \tilde{e}_{h,M} + \tilde{\eta}_{h,M} \| \leq C_{\text{control}} Tol_M.
\]

If it is true, the overall error \( E_h(T) = V_h(T) - R_h u(T) = e_h(T) + \eta_h(T) \) is considered small enough relative to the chosen tolerance and \( V_{h,M} \) is accepted. Otherwise, the whole computation is redone with the (already) adjusted tolerances (52) and an improved spatial resolution.

In the uniform case, we use the new mesh size computed from (17) with \( Tol = (1 - C_T) Tol_M \). To check the convergence behaviour in space and therefore also the quality of the approximation of the spatial truncation error, we additionally compute the numerically observed order

\[
q_{\text{num}} = \log \left( \frac{\| \tilde{\eta}_{h,M} \|}{\| \tilde{\eta}_{h_{\text{new}},M} \|} \right) / \log \left( \frac{h}{h_{\text{new}}} \right) .
\]
Step 0 Choose global tolerances $G\text{tol}_A$ and $G\text{tol}_R$.
Choose $C_T$, $C_{\text{control}}$, $h_0$, $q$, and $\tau_0$.
Set local tolerances $T\text{ol}_A = G\text{tol}_A$ and $T\text{ol}_R = G\text{tol}_R$.
Set $h = h_0$.

Step 1 Run numerical schemes to compute $V_{h,M}$, $\tilde{e}_{h,M}$, $\tilde{\eta}_{h,M}$.
Compute $T\text{ol}_M = G\text{tol}_A + G\text{tol}_R \|V_{h,M}\|$.

Step 2 IF $\|\tilde{e}_{h,M}\| \leq C_{\text{control}} T\text{ol}_M$ GOTO Step 3.
ELSE set $f_{\text{ac}} = C_T T\text{ol}_M / \|\tilde{e}_{h,M}\|$, $T\text{ol}_A = T\text{ol}_A \cdot f_{\text{ac}}$, $T\text{ol}_R = T\text{ol}_R \cdot f_{\text{ac}}$ and GOTO Step 1.

Step 3 IF $\|\tilde{e}_{h,M} + \tilde{\eta}_{h,M}\| \leq C_{\text{control}} T\text{ol}_M$ GOTO Step 4.
ELSE set $h = \sqrt{(1 - C_T) T\text{ol}_M / \|\tilde{\eta}_{h,M}\|}$ and GOTO Step 1.

Step 4 IF $h \neq h_0$ compute $q_{\text{num}}$.
ELSE set $h = 2h$, run numerical schemes again and compute then $q_{\text{num}}$.
IF $q_{\text{num}} \approx q$ accept fine grid solution and STOP.
ELSE set $h_0 = 2h_0, h = h_0$ and GOTO Step 1.

| Step | Control Algorithm with Uniform Refinement in Space |
|------|--------------------------------------------------|
| 0    | Choose global tolerances $G\text{tol}_A$ and $G\text{tol}_R$. Choose $C_T$, $C_{\text{control}}$, $h_0$, $q$, and $\tau_0$. Set local tolerances $T\text{ol}_A = G\text{tol}_A$ and $T\text{ol}_R = G\text{tol}_R$. Set $h = h_0$. |
| 1    | Run numerical schemes to compute $V_{h,M}$, $\tilde{e}_{h,M}$, $\tilde{\eta}_{h,M}$. Compute $T\text{ol}_M = G\text{tol}_A + G\text{tol}_R \|V_{h,M}\|$. |
| 2    | IF $\|\tilde{e}_{h,M}\| \leq C_T C_{\text{control}} T\text{ol}_M$ GOTO Step 3. ELSE set $f_{\text{ac}} = C_T T\text{ol}_M / \|\tilde{e}_{h,M}\|$, $T\text{ol}_A = T\text{ol}_A \cdot f_{\text{ac}}$, $T\text{ol}_R = T\text{ol}_R \cdot f_{\text{ac}}$ and GOTO Step 1. |
| 3    | IF $\|\tilde{e}_{h,M} + \tilde{\eta}_{h,M}\| \leq C_{\text{control}} T\text{ol}_M$ GOTO Step 4. ELSE set $h = \sqrt{(1 - C_T) T\text{ol}_M / \|\tilde{\eta}_{h,M}\|}$ and GOTO Step 1. |
| 4    | IF $h \neq h_0$ compute $q_{\text{num}}$. ELSE set $h = 2h$, run numerical schemes again and compute then $q_{\text{num}}$. IF $q_{\text{num}} \approx q$ accept fine grid solution and STOP. ELSE set $h_0 = 2h_0, h = h_0$ and GOTO Step 1. |

Table 1. Algorithmic structure of the overall control strategy where uniform refinement in space is used.

If $q_{\text{num}}$ computed for the final run is not close to the expected value $q$ used for our Richardson extrapolation, we reason that the approximation of the spatial truncation errors has failed due to a dominating global time error, which happens, e.g., if the initial spatial mesh is already too fine. Consequently, we coarsen the initial mesh by a factor two and start again. If the control approach stops without a mesh refinement, we perform an additional control run on the coarse mesh and compute $q_{\text{num}}$ from (54) with $h_{\text{new}} = 2h$. It turns out that this simple strategy works quite robustly, provided that the meshes used are able to resolve the basic behaviour of the solution. The algorithmic structure of our control strategy with uniform refinement in space is given in Table 1.

In the adaptive case, we choose new local tolerances

$$T\text{ol}_A^\alpha = T\text{ol}_A^\alpha \cdot f_{\text{ac}}, \quad T\text{ol}_R^\alpha = T\text{ol}_R^\alpha \cdot f_{\text{ac}}, \quad f_{\text{ac}} = (1 - C_T) T\text{ol}_M / \|\tilde{\eta}_{h,M}\|,$$

and the whole computation is redone over the interval $[0, T]$. Based on tolerance proportionality, reducing the local truncation error with the factor $f_{\text{ac}}$ will reduce $\eta_h(T)$ by $f_{\text{ac}}$. In Table 2 the algorithmic structure of our control strategy with adaptive refinement in space is displayed. Note that now the index $h$ refers to a sequence of spatial meshes adapted at each time point $t_n$.

Summarizing, the first check (51) and the possibly second control computation serve to significantly reduce the global time error. This enables us to make use of the approximation (38) for the spatial truncation error, which otherwise could not be trusted. The second step based on suitable spatial mesh improvement attempts to bring the overall error down to the imposed tolerance. Using the sum of the approximate global time and spatial error inside the norm in (53), we take advantage of favourable effects of error cancellation. These two steps are successively repeated until the second check is successful. Additionally, if uniform...
Step 0. Choose global tolerances $GTol_A$ and $GTol_R$.
Choose $C_T$, $C_{control}$, $C_\alpha$, $q$, and $\tau_0$.
Set local tolerances $Tol_A = GTol_A$, $Tol_R = GTol_R$, $Tol_A^\alpha = C_\alpha GTol_A$, and $Tol_R^\alpha = C_\alpha GTol_R$.
Choose initial spatial mesh.

Step 1. Run numerical schemes to compute $V_{h,M}, \tilde{e}_{h,M}, \tilde{\eta}_{h,M}$.
Compute $Tol_M = GTol_A + GTol_R \| V_{h,M} \|$.

Step 2. IF $\| \tilde{e}_{h,M} \| \leq C_T C_{control} Tol_M$ GOTO Step 3.
ELSE set $fac = C_T Tol_M / \| \tilde{e}_{h,M} \|$, $Tol_A = Tol_A \cdot fac$, $Tol_R = Tol_R \cdot fac$ and GOTO Step 1.

Step 3. IF $\| \tilde{e}_{h,M} + \tilde{\eta}_{h,M} \| \leq C_{control} Tol_M$ accept solution and STOP.
ELSE set $fac = (1 - C_T) Tol_M / \| \tilde{\eta}_{h,M} \|$, $Tol_A^\alpha = Tol_A^\alpha \cdot fac$, $Tol_R^\alpha = Tol_R^\alpha \cdot fac$ and GOTO Step 1.

Table 2. Algorithmic structure of the overall control strategy where adaptive refinement in space is used.

Mesh refinement is used we take into account the numerically observed order in space to assess the approximation of the spatial truncation error.

7. Numerical Illustrations

To illustrate the performance of the global error estimators and the control strategy, we consider three test problems: (i) the highly stable heat equation with nonhomogeneous Neumann boundary conditions [1], (ii) the nonlinear convection-dominated Burgers’ equation [1, 6], and (iii) the Allen-Cahn equation modelling a diffusion-reaction problem [4]. Analytic solutions are known for all three problems. Uniform spatial refinement is studied for all three test cases. For the Allen-Cahn problem, these results are compared to those obtained with adaptive refinement.

We set $GTol_A = GTol_R = GTol$ for $GTol = 10^{-l}, l = 2, \ldots, 7$ and start with one and the same initial step size $\tau_0 = 10^{-5}$. Equally spaced meshes of 25, 51, 103, 207, 415, 831, and 1663 points are used as initial mesh. The control parameters introduced above for the control rules are $C_T = 1/3$, $C_{control} = 1.2$, and $C_\alpha = 10$. All runs were performed, but for convenience we only select a representative set of them for our presentation.

We define the estimated global error $\hat{E}_{h,M} = \hat{e}_{h,M} + \hat{\eta}_{h,M}$ at time $t = T$ and set indicators $\Theta_{est} = \| \hat{E}_{h,M} \| / \| E_h(T) \|$ for the ratio of the estimated global error and the true global error, and $\Theta_{ctr} = Tol_M / \| E_h(T) \|$ for the ratio of the desired tolerance and the true global error. Thus, $\Theta_{ctr} \geq 1/C_{control} = 5/6$ indicates control of the true global error.

The tables of results contain the following quantities, $Tol = Tol_A = Tol_R$ from (52), $Tol^\alpha = Tol_A^\alpha = Tol_R^\alpha$ from (55), $Tol_M = GTol (1 + \| V_{h,M} \|)$ from (51), the estimated global error $\hat{E}_{h,M}$, the estimated time error $\hat{e}_{h,M}$, and the estimated spatial truncation error $\hat{\eta}_{h,M}$. Note that we always start with $Tol = GTol$ in the first run. The ratios $\Theta_{est}$ and $\Theta_{ctr}$ serve to illustrate the quality of the global error estimation and the control. If uniform refinement
in space is applied, the numerically observed order $q_{num}$ for the spatial error is given. It will be clear from the tables of results whether a tolerance-adapted run to control the global time error, a spatial mesh adaption step or an additional control run on a coarser grid was necessary. Especially, the latter is marked by a dashed line.

| $T ol$ | $N$ | $T ol_M$ | $\| \tilde{E}_{h,M} \|$ | $\| \tilde{e}_{h,M} \|$ | $\| \tilde{\eta}_{h,M} \|$ | $\Theta_{est}$ | $\Theta_{ctr}$ | $q_{num}$ |
|---|---|---|---|---|---|---|---|---|
| 1.00e-2 | 25 | 1.10e-2 | 7.14e-4 | 1.16e-4 | 8.20e-4 | 0.99 | 15.27 |
| 1.00e-2 | 13 | 1.10e-2 | 3.27e-5 | 1.24e-4 | 3.38e-3 | 0.99 | 3.32 | 2.04 |
| 1.00e-3 | 51 | 1.10e-3 | 1.68e-4 | 1.97e-5 | 1.86e-4 | 1.00 | 6.51 |
| 1.00e-3 | 25 | 1.10e-3 | 8.04e-4 | 2.03e-5 | 8.22e-4 | 1.00 | 1.36 | 2.02 |
| 1.00e-4 | 103 | 1.10e-4 | 4.27e-5 | 2.01e-6 | 4.44e-5 | 1.00 | 2.57 |
| 1.00e-4 | 51 | 1.10e-4 | 1.85e-4 | 1.96e-6 | 1.86e-4 | 1.00 | 0.59 | 2.01 |
| 1.00e-5 | 207 | 1.10e-5 | 1.07e-5 | 1.08e-7 | 1.08e-5 | 1.00 | 1.03 |
| 1.00e-5 | 103 | 1.10e-5 | 4.43e-5 | 4.44e-7 | 4.44e-5 | 1.00 | 0.25 | 2.01 |
| 1.00e-6 | 415 | 1.10e-6 | 2.67e-6 | 2.68e-6 | 2.68e-6 | 1.00 | 0.41 |
| 1.00e-6 | 795 | 1.10e-6 | 7.14e-7 | 7.28e-7 | 7.28e-7 | 1.00 | 1.54 | 2.00 |
| 1.00e-7 | 25 | 1.10e-7 | 8.24e-4 | 8.24e-4 | 8.24e-4 | 1.00 | 0.00 |
| 1.00e-7 | 2759 | 1.10e-7 | 5.91e-8 | 6.03e-8 | 6.03e-8 | 1.00 | 1.86 | 2.01 |
| 1.00e-7 | 1663 | 1.10e-7 | 1.65e-7 | 1.66e-7 | 1.66e-7 | 1.00 | 0.67 |
| 1.00e-7 | 2505 | 1.10e-7 | 7.20e-8 | 7.31e-8 | 7.31e-8 | 1.00 | 1.53 | 2.00 |

Table 3. Selected data for the heat equation with Neumann boundary conditions. Uniform refinement in space is used.

7.1. **Heat equation with Neumann boundary conditions.** This heat equation provides an example with inhomogeneous Neumann boundary conditions:

$$\partial_t u = \partial_{xx} u, \quad 0 < x < 1.0, \quad 0 < t \leq T = 0.2,$$

and boundary conditions $\partial_x u = \pi e^{-\pi^2 t} \cos(\pi x)$ at $x = 0$ and $x = 1$. The initial condition is consistent with the analytic solution $u(x, t) = e^{-\pi^2 t} \sin(\pi x)$. Although the solution is very stable, it is not easy to provide good error estimates as stated in [1, 6].

To approximate the inhomogeneous Neumann boundary conditions we introduce artificial mesh points $x_{-1} = -h$ and $x_{N+2} = 1+h$, discretize $\partial_x u(0)$ and $\partial_x u(1)$ by second order central differences, and use the approximate differential equation at the boundary to eliminate the artificial solution values. In consequence, we have second order in all mesh points and $q = 2$ works also fine for interpolating the estimated spatial truncation error at the boundary.

Due to the high stability of the problem the global time errors are much smaller than imposed local tolerances. So, control of the global time error is redundant here and control runs were only carried out in case of insufficient spatial resolutions. Table 3 shows results for various tolerances and initial meshes. The global error estimation and control appear to work very well for this problem, where the influence of the initial mesh points is less strong. This holds also for other combinations of tolerances and initial meshes. Note the high quality of the estimator $\tilde{E}_{h,M}$ (and therefore also of $\tilde{\eta}_{h,M}$), showing that the derivative boundary condition is well resolved within the Richardson extrapolation. For the runs with tolerances
$GTol = 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$, the order of the spatial convergence was successfully checked with a second run on the coarse mesh, that is, we can trust the first run.

### Table 4. Selected data for Burgers’ equation with 51 initial mesh points.

| $Tol$     | $N$  | $Tol_M$ | $||E_h,M||$ | $||\tilde{e}_h,M||$ | $||\eta_h,M||$ | $\Theta_{est}$ | $\Theta_{ctr}$ | $q_{num}$ |
|-----------|------|---------|-------------|--------------------|----------------|----------------|----------------|----------|
| $1.00e-2$ | 51   | 1.93e-2 | 4.30e-3     | 1.86e-3            | 2.86e-3        | 1.08           | 4.87           |          |
| $1.00e-2$ | 25   | 1.93e-2 | 1.29e-2     | 2.21e-2            | 1.14e-2        | 0.99           | 1.48           | 2.00     |
| $1.00e-3$ | 51   | 1.93e-3 | 2.83e-3     | 1.54e-4            | 2.74e-3        | 0.99           | 0.68           |          |
| $1.00e-3$ | 75   | 1.93e-3 | 1.36e-3     | 1.48e-4            | 1.28e-3        | 1.00           | 1.42           | 2.00     |
| $1.00e-4$ | 51   | 1.93e-4 | 2.73e-3     | 1.09e-5            | 2.73e-3        | 0.98           | 0.07           |          |
| $1.00e-4$ | 239  | 1.94e-4 | 1.32e-4     | 1.05e-5            | 1.27e-4        | 1.00           | 1.46           | 2.00     |
| $1.00e-5$ | 51   | 1.93e-5 | 2.73e-3     | 1.08e-6            | 2.73e-3        | 0.98           | 0.01           |          |
| $1.00e-5$ | 757  | 1.94e-5 | 1.32e-5     | 1.02e-6            | 1.27e-5        | 1.00           | 1.47           | 2.00     |
| $1.00e-6$ | 51   | 1.93e-6 | 2.73e-3     | 1.08e-7            | 2.73e-3        | 0.98           | 0.00           |          |
| $1.00e-6$ | 2391 | 1.94e-6 | 1.32e-6     | 9.29e-8            | 1.28e-6        | 1.00           | 1.47           | 2.00     |
| $1.00e-7$ | 51   | 1.93e-7 | 2.73e-3     | 1.10e-8            | 2.73e-3        | 0.98           | 0.00           |          |
| $1.00e-7$ | 7563 | 1.94e-7 | 1.31e-7     | 8.57e-9            | 1.28e-7        | 1.00           | 1.47           | 2.00     |

7.2. Burgers’ equation. The second problem is the nonlinear Burgers’ equation

$$\partial_t u = \varepsilon \partial_{xx} u - u \partial_x u, \quad 0 < x < 1.0, \quad 0 < t \leq T = 1.0,$$

where $\varepsilon = 0.015$ is used in the experiments. Dirichlet boundary conditions and initial conditions are consistent with the analytic solution defined by

$$u(x, t) = r_1 + 5r_2 + 10r_3 \over 10(r_1 + r_2 + r_3),$$

where $r_1(x) = e^{0.45x/\varepsilon}$, $r_2(t, x) = e^{0.01(10 + 6x + 25x)/\varepsilon}$, and $r_3(t) = e^{0.025(6.5 + 9.9t)/\varepsilon}$.

In Table 4 we present results for all tolerances used and the 51-point initial mesh. The use of a relatively coarse mesh at the beginning is the natural choice in practice. No adaptation in time is necessary, which is mainly due to the small first time step and the maximum factor 1.5 which is allowed in (43) for a step size enlargement. For the tolerance $GTol = 10^{-2}$, the numerical solution is accepted since the corresponding control run shows $q_{num} \approx 2$, the expected value. Remarkably excellent estimates are obtained for higher tolerances. Here, control is always achieved after one spatial mesh improvement.

7.3. The Allen-Cahn equation. The third problem is the bi-stable Allen-Cahn equation which is defined by

$$\partial_t u = 10^{-2} \partial_{xx} u + 100u (1 - u^2), \quad 0 < x < 2.5, \quad 0 < t \leq T = 0.5,$$

with the initial function and Dirichlet boundary values taken from the exact wave front solution $u(x, t) = (1 + e^{\lambda(x-\alpha t)})^{-1}$, $\lambda = 50 \sqrt{2}$, $\alpha = 1.5 \sqrt{2}$. This problem was also used in [4].

First we apply uniform refinement in space. Table 5 reveals a high quality of the global error estimation and also the control process works quite well. Let us pick one exemplary run.
| $Tol$ | $N$ | $Tol_M$ | $\| \tilde{E}_{h,M} \|$ | $\| \tilde{e}_{h,M} \|$ | $\| \tilde{\eta}_{h,M} \|$ | $\Theta_{est}$ | $\Theta_{ctr}$ | $q_{num}$ |
|-------|-----|--------|-------------------|------------------|-------------------|-------------|-------------|--------|
| 1.00e-2 | 103 | 2.05e-2 | 1.84e-0 | 1.45e-1 | 1.98e-0 | 9.89 | 0.11 |
| 4.69e-4 | 103 | 2.05e-2 | 5.78e-1 | 1.26e-3 | 5.79e-1 | 2.69 | 0.10 |
| 4.69e-4 | 677 | 2.02e-2 | 6.04e-3 | 1.11e-3 | 7.15e-3 | 1.19 | 3.98 |
| $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ |
| 1.00e-2 | 415 | 2.02e-2 | 7.69e-2 | 1.44e-1 | 6.73e-2 | 3.05 | 0.80 |
| 4.66e-4 | 415 | 2.02e-2 | 1.86e-2 | 1.11e-3 | 1.97e-2 | 1.23 | 1.34 |
| 4.66e-4 | 207 | 2.03e-2 | 9.17e-2 | 1.15e-3 | 9.29e-2 | 1.47 | 0.32 |
| $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ |
| 1.00e-3 | 207 | 2.03e-3 | 9.82e-2 | 2.97e-3 | 1.01e-1 | 1.60 | 0.03 |
| 2.27e-4 | 207 | 2.03e-3 | 8.80e-2 | 4.93e-4 | 8.85e-2 | 1.39 | 0.03 |
| 2.27e-4 | 1683 | 2.02e-3 | 6.14e-4 | 4.71e-4 | 1.19 e-3 | 1.19 | 3.98 |
| $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ |
| 1.00e-3 | 415 | 2.02e-3 | 2.26e-3 | 2.87e-3 | 5.12e-3 | 1.33 | 1.19 |
| 6.35e-5 | 415 | 2.02e-3 | 4.01e-3 | 4.91e-4 | 4.50e-3 | 1.23 | 1.34 |
| 6.35e-5 | 1521 | 2.02e-3 | 1.34e-3 | 1.85e-3 | 3.05e-2 | 3.05 | 0.80 |
| $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ |
| 1.00e-4 | 207 | 2.02e-4 | 1.86e-3 | 1.08e-3 | 1.11 | 2.68 |
| 3.63e-5 | 207 | 2.02e-4 | 9.82e-2 | 2.97e-3 | 1.01e-1 | 1.60 | 0.03 |
| 3.63e-5 | 4643 | 2.02e-4 | 7.30e-5 | 6.14e-5 | 1.34e-3 | 1.19 |
| $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ | $\times$ |

**Table 5.** Selected data for the Allen-Cahn problem. Uniform refinement in space is used.

| $Tol$ | $Tol^a$ | $N_M$ | $Tol_M$ | $\| E_{h,M} \|$ | $\| \tilde{e}_{h,M} \|$ | $\| \tilde{\eta}_{h,M} \|$ | $\Theta_{est}$ | $\Theta_{ctr}$ |
|-------|--------|-----|--------|-------------------|------------------|-------------------|-------------|-------------|
| 1.00e-2 | 1.00e-1 | 245 | 2.01e-2 | 1.05e-1 | 1.39e-1 | 3.42e-2 | 3.21 | 0.61 |
| 4.81e-4 | 1.00e-1 | 247 | 2.01e-2 | 8.54e-3 | 1.13e-3 | 9.67e-3 | 1.26 | 2.98 |
| 1.00e-3 | 1.00e-2 | 483 | 2.01e-3 | 1.26e-3 | 2.86e-3 | 1.59e-3 | 1.26 | 2.98 |
| 2.35e-4 | 1.00e-2 | 481 | 2.01e-3 | 9.72e-4 | 4.84e-4 | 1.46e-3 | 1.26 | 2.98 |
| 1.00e-4 | 1.00e-3 | 1839 | 2.01e-4 | 9.08e-5 | 1.85e-4 | 9.45e-5 | 1.63 | 3.62 |
| 3.62e-5 | 1.00e-3 | 1839 | 2.01e-4 | 5.49e-5 | 6.06e-5 | 1.16e-4 | 0.92 | 3.36 |
| 1.00e-4 | 1.00e-2 | 483 | 2.01e-4 | 1.23e-3 | 1.85e-4 | 1.41e-3 | 1.08 | 0.18 |
| 3.62e-5 | 1.00e-2 | 483 | 2.01e-4 | 1.32e-3 | 6.09e-5 | 1.38e-3 | 1.06 | 0.16 |
| 3.62e-5 | 9.68e-4 | 1809 | 2.01e-4 | 5.48e-5 | 6.07e-5 | 1.15e-4 | 0.92 | 3.36 |
| 1.00e-4 | 1.00e-1 | 243 | 2.01e-4 | 8.66e-3 | 1.84e-4 | 8.84e-3 | 1.15 | 0.03 |
| 3.62e-5 | 1.00e-1 | 243 | 2.01e-4 | 8.55e-3 | 6.08e-5 | 8.61e-3 | 1.12 | 0.03 |
| 3.62e-5 | 1.55e-3 | 1809 | 2.01e-4 | 5.68e-5 | 6.06e-5 | 1.17e-4 | 0.92 | 3.25 |

**Table 6.** Selected data for the Allen-Cahn problem. Adaptive refinement in space is used.

To explain the overall control strategy in more detail. Starting with $GTol = Tol = 10^{-3}$ and 831 mesh points, which corresponds to the fourth simulation, the numerical scheme delivers global error estimates $\| \tilde{e}_{h,M} \| = 2.87 \times 10^{-3}$ and $\| \tilde{\eta}_{h,M} \| = 5.12 \times 10^{-3}$ for the time and spatial error of the approximate solution $V_{h,M}$ at the final time $t_M = T$. The first check for the time error estimate $\| \tilde{e}_{h,M} \| \leq C_T C_{control} Tol_M = 8.08 \times 10^{-4}$ fails and we adjust the local tolerances by a factor $fac = C_T Tol_M / \| \tilde{e}_{h,M} \| = 2.35 \times 10^{-1}$, which yields the new $Tol = 2.35 \times 10^{-4}$. The computation is then redone. Due to the tolerance proportionality, in
the second run the time error is significantly reduced and the inequality $\|\tilde{e}_{h,M}\| \leq 8.08 \times 10^{-4}$ is now valid. We proceed with checking $\|\tilde{E}_{h,M}\| \leq C_{\text{control}}\text{Tol}_M = 2.42 \times 10^{-3}$, which is still not true. From \[ (47) \], we compute a new number of spatial mesh points $N = 1521$. Finally, the third run is successful and with the numerically observed spatial order $q_{\text{num}} = 2.02$ the numerical solution is accepted.

The ratios for $\Theta_{\text{est}} = \|\tilde{E}_{h,M}\|/\|E_h(T)\|$ lie between 1.04 and 1.23, after the control runs. Control of the global error, that is $\|E_h(T)\| \leq C_{\text{control}}\text{Tol}_M$, is in general achieved after two steps (one step to adjust the time grid and one step to control the space discretization), whereas the efficiency index $\Theta_{\text{ctr}} = \text{Tol}_M/\|E_h(T)\|$ is close to three. This results from a systematic cancellation effect between the global time and spatial error, which is not taken into account when computing $h_{\text{new}}$ from \[ (47) \].

Next we consider locally adaptive spatial grid enhancement instead of globally uniform adaptation. Within each time step the grid is adapted by refinement and coarsening, based on an equidistribution principle, until $A_n \leq \text{Tol}_n = \text{Tol}^n(1+\|V_{h,n}\|)$ holds. This yields a sequence of non-uniform meshes. Let $N_M$ denote the number of adaptive grids points obtained at the final time $T$. The first three runs in Table 6 correspond to our standard setting $C_\alpha = 10$, i.e., $\text{Tol}_n = 10\text{Tol}$. In this case, after adjusting the local tolerances for the time integration no further run with higher tolerances in space is necessary. To demonstrate the robustness of the algorithm, we select two additional runs with $C_\alpha = 10^l$, $l = 2, 3$, for $G\text{Tol} = 10^{-4}$. In both cases, coarser meshes are used at the beginning and a second control run has to be done to decrease the spatial discretization error. The resulting adaptive spatial meshes are comparable. Control of the global error is always achieved. The estimation process works again quite well. Compared to the uniform case, significantly less spatial degrees of freedoms are needed to reach the desired tolerances. The reduction rate varies between 40% and 70%.

8. Summary

We have developed an error control strategy for finite difference solutions of parabolic equations, involving both temporal and spatial discretization errors. The global time error strategy discussed in \[ 1 \] appears to provide an excellent starting point for the development of such an algorithm. The classical ODE approach based on the first variational equation and the principle of tolerance proportionality is combined with an efficient estimation of the spatial error and mesh adaptation to control the overall global error. Two approaches have been presented to handle spatial mesh improvement: (i) globally uniform refinement and (ii) local refinement and coarsening based on an equidistribution principle. Inspired by \[ 1 \], we have used Richardson extrapolation to approximate the spatial truncation error within the method of lines. Our control strategy aims at balancing the spatial and temporal discretization error in order to achieve an accuracy imposed by the user.

The key ingredients are: (i) linearised error transport equations equipped with sufficiently accurate defects to approximate the global time error and global spatial error and (ii) uniform or adaptive mesh refinement and local error control in time based on tolerance proportionality to achieve global error control. For illustration of the performance and effectiveness of our approach, we have implemented second-order finite differences in one space dimension and the example integrator ROS3P \[ 3 \]. On the basis of three different test problems we could observe that our approach is very reliable, both with respect to estimation and control.
9. Acknowledgement

This work has been supported in part by the German Research Foundation (DFG) by the grant SFB568.

References

[1] M. Berzins (1988), Global error estimation in the method of lines for parabolic equations, SIAM J. Sci. Stat. Comput. 9, pp. 687-703.
[2] J. Lang (2000), Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems. Theory, Algorithm and Applications, Lecture Notes in Computational Science and Engineering, Vol. 16, Springer.
[3] J. Lang, J.G. Verwer (2001), ROS3P – An accurate third-order Rosenbrock solver designed for parabolic problems, BIT 41, pp. 731-738.
[4] J. Lang, J.G. Verwer (2007), On global error estimation and control for initial value problems, SIAM J. Sci. Comput. 29, pp. 1460-1475.
[5] S. Larsson, V. Thomée (2005), Partial Differential Equations with Numerical Methods, Texts in Applied Mathematics, Vol. 45, 2nd printing, Springer.
[6] L. Lawson, M. Berzins, P.M. Dew (1991), Balancing space and time errors in the method of lines for parabolic equations, SIAM J. Sci. Stat. Comput. 12, pp. 573-594.
[7] L.F. Shampine (1994), Numerical Solution of Ordinary Differential Equations, Chapman & Hall, New York.
[8] R.D. Skeel (1986), Thirteen ways to estimate global error, Numer. Math. 48, pp. 1-20.
[9] J.W. Thomas (1995), Numerical Partial Differential Equations. Finite Difference Methods, Text in Applied Mathematics 22, Springer.

Kristian Debrabant, Department of Mathematics, Technische Universität Darmstadt, Dolivostr. 15, D-64293 Darmstadt, Germany
E-mail address: debrabant@mathematik.tu-darmstadt.de

Jens Lang, Department of Mathematics, Technische Universität Darmstadt, Dolivostr. 15, D-64293 Darmstadt, Germany
E-mail address: lang@mathematik.tu-darmstadt.de