An efficient way to avoid the order reduction of linearly implicit Runge-Kutta methods for nonlinear IBVP's

M. P. Calvo, J. de Frutos, and J. Novo

Abstract. In the present paper we show that the strategy proposed in [7] to avoid the order reduction of Runge-Kutta methods when integrating linear initial boundary value problems can be extended to also avoid the order reduction in nonlinear cases. Furthermore, we see that if the Runge-Kutta method is replaced by an appropriate linearly implicit Runge-Kutta method, this strategy is interesting not only from a theoretical point of view, but also provides an efficient procedure to time integrate initial boundary value problems.

1. Introduction

Let us consider an abstract initial boundary value problem

$$\left\{\begin{array}{ll} u'(t) = \widetilde{A}u(t) + f(t), & 0 \le t \le T, \\ u(0) = u_0, & \\ \partial u(t) = g(t), & 0 \le t \le T, \end{array}\right. \tag{1.1}$$

where $\widetilde{A} : D(\widetilde{A}) \subset X \to X$ and $\partial : D(\widetilde{A}) \subset X \to Y$ are linear operators, X and Y are Banach spaces, $u_0 \in X$, $f : [0,T] \to X$ and $g : [0,T] \to Y$. Many evolutionary partial differential equations of practical interest can be written in this abstract form [14]. It is well known that when (1.1) is integrated in time with a Runge-Kutta method of classical order p and stage order q [10], the order of convergence is $\min\{p, q+\theta\}$ with $0 \le \theta < 2$ [5]. This is the so called *order reduction phenomenom.*

Different remedies to avoid the order reduction of Runge-Kutta methods when applied to the time integration of initial boundary value problems have been proposed in the literature [1, 2, 3, 4, 5, 7, 9, 11, 12, 15, 16, 18]. The aim of the

1991 *Mathematics Subject Classification.* Primary 65M12, 65M20.

Key words and phrases. Runge-Kutta, order reduction, linearly implicit methods.

This research has been supported by DGI-MCYT under project BFM2001-2138 cofinanciated by FEDER funds and by Junta de Castilla y León under project VA112/02. Research completed while the second author was visiting professor at GERAD and HEC (Montréal), under grant SEEU PR2002-0223.

present paper is to show that the strategy proposed in [7] to deal with linear problems can be applied to also avoid the order reduction of linearly implicit Runge-Kutta methods when time integrating nonlinear problems.

The basic idea of the strategy proposed in [7] is as follows. The solution of (1.1) is decomposed as $u(t) = v(t) + u^*(t)$, where $v(t)$ can be computed directly in terms of the data and $u^*(t)$ is the solution of a suitable initial value problem for which there is no order reduction. In order to extend this idea to deal with nonlinear problems, we need to describe more precisely how these functions are constructed in [7].

For the sake of simplicity, throughout the paper we assume that the problem

$$\left\{ \begin{array}{l} \widetilde{A}x = 0, \\ \partial x = y \end{array} \right. \tag{1.2}$$

has a unique solution and we denote it by Ey. The technical hypothesis to be fulfilled by $\widetilde{A}$ and ∂ are described in Section 2 of [7]. Set $D(A) = \ker(\partial) = \{ x \in D(\widetilde{A}) \,/\, \partial x = 0 \}$ and let us consider the restriction $A = \widetilde{A}|_{D(A)} : D(A) \subset X \to X$ which, under these hypothesis, is the infinitesimal generator of a C_0-semigroup in X. Let us also assume that the solution u of (1.1) belongs to $C^{p+1}([0,T], D(\widetilde{A}^r))$ for some $1 \le r \le p-q$. (Here p and q denote the order and the stage order of the Runge-Kutta method, respectively). We want to stress that, in practice, this is a natural requirement since it only demands regularity of u both, in time and space, but it does not impose any restriction on the boundary values of u. The correcting term is defined in [7] as

$$v_r(t) = \sum_{s=0}^{r-1} A^{-s} E \partial \widetilde{A}^s u(t), \qquad 0 \le t \le T. \tag{1.3}$$

Then $u^{*,r}(t) = u(t) - v_r(t)$ is the solution of the initial value problem

$$\left\{ \begin{array}{ll} (u^{*,r})'(t) = Au^{*,r}(t) + f^{*,r}(t), & 0 \le t \le T, \\ u^{*,r}(0) = u_0 - v_r(0), \end{array} \right. \tag{1.4}$$

where

$$f^{*,r}(t) = f(t) - \sum_{s=0}^{r-1} A^{-s} E \partial \widetilde{A}^s f(t) - A^{-(r-1)} E \left[g^{(r)}(t) - \sum_{j=0}^{r-1} \partial \widetilde{A}^{r-j-1} f^{(j)}(t) \right].$$

We want to stress that, as shown in [7],

$$\partial \widetilde{A}^s u(t) = g^{(s)}(t) - \sum_{j=0}^{s-1} \partial \widetilde{A}^{s-j-1} f^{(j)}(t), \qquad 1 \le s \le r-1.$$

Therefore, $v_r(t)$ and $f^{*,r}(t)$ can be computed in terms of f and g and their derivatives. This will not happen when dealing with nonlinear problems.

Denoting by $u_n^{*,r}$ the Runge-Kutta approximation to the solution of (1.4) at time level t_n, $0 \le n \le N$ and setting $\hat{u}_n = u_n^{*,r} + v_r(t_n)$ (then $\hat{u}_n$ is the numerical approximation to the solution of (1.1) at time level t_n), in [7] the following theorem is proved:

Theorem 1.1. *Let $u \in C^{p+1}([0,T], D(\widetilde{A}^r))$, $r = p - q$, be the solution of the initial boundary value problem (1.1). Assume also that $f \in C^p([0,T], X)$ and $g \in C^{p+1}([0,T], Y)$. Let $v_r : [0,T] \to X$ be the correcting term defined in (1.3). Then*

$$\|u(t_n) - \hat{u}_n\| = \|u^{*,r}(t_n) - u_n^{*,r}\| \le C\rho_n \sum_{m=1}^{n} k_{m-1}^{p+1} I_m(u),$$

where $C > 0$ is a constant and

$$\begin{aligned} \rho_n &= \| \prod_{m=1}^{n} r(k_{m-1}A)\|, \\ I_m(u) &= \sup_{q+1 \le l \le p+1} \|u^{(l)}\|_{L^\infty([t_{m-1},t_m], D(\widetilde{A}^r))}. \end{aligned}$$

Here $r(z)$ is the stability function of the Runge-Kutta method, and the step sizes k_m, $0 \le m \le n-1$, are small enough to ensure that the operators $r(k_m A)$ are well defined in X. The constant C depends on T, A and the Runge-Kutta method but it is independent of u, f and g.

Observe that $r = 1$ in (1.3) corresponds to the usual correction $v_1(t) = Eg(t)$ mentioned in [16] to convert problem (1.1) into an initial boundary value problem with homogeneous boundary conditions, which is less affected than (1.1) by the order reduction [5], [13]. For $r = 2$,

$$v_2(t) = Eg(t) + A^{-1}E[g'(t) - \partial f(t)] \tag{1.5}$$

and

$$f^{*,2}(t) = f(t) - E\partial f(t) - A^{-1}E[g''(t) - \partial f'(t)]. \tag{1.6}$$

In Section 2 we present a nonlinear initial boundary value problem which exhibits order reduction when is time integrated using a third order linearly implicit Runge-Kutta method. Section 3 is devoted to adapt the strategy of [7] to deal with nonlinear problems. In Section 4 we show how this strategy can be efficiently implemented and, finally, in Section 5 numerical results are provided.

2. A nonlinear problem with order reduction

Let us consider the one dimensional forced Burgers' equation

$$u_t = u_{xx} - uu_x + f(t,x), \qquad -1 \le x \le 1, \qquad t > 0, \tag{2.1}$$

subject to homogeneous Dirichlet boundary conditions and with initial data

$$u_0(x) = \begin{cases} 1 + x + \dfrac{x^2}{2} + \dfrac{x^3}{6} - \dfrac{x^4}{3} & \text{if } x \in [-1,0], \\[2ex] 1 + x + \dfrac{x^2}{2} + \dfrac{x^3}{6} - \dfrac{8x^4}{3} & \text{if } x \in [0,1]. \end{cases}$$

The forcing term f is defined in order to have $u(t,x) = u_0(x)/(1+t)$ as the exact solution of the problem. We chose homogeneous boundary conditions since, in practice, before starting the numerical integration, the original problem is transformed into an initial boundary value problem satisfying homogeneous boundary conditions.

Let us numerically integrate equation (2.1). For the spatial discretization we consider a Chebyshev spectral Galerkin method [8] using the basis functions proposed in [17] and evaluating the nonlinear terms using collocation and Fast Fourier Transform techniques. The resulting semidiscretized system has the form

$$\mathbf{y}' = L\mathbf{y} + N(t,\mathbf{y}), \tag{2.2}$$

where matrix L stems from the spatial discretization of the diffusive term and $N(t,\mathbf{y})$ arises from the discretization of nonlinear and source terms. The main difficulty in dealing with (2.2) is that the use of explicit integrators is usually inefficient because the system becomes stiffer as the spatial mesh is refined. On the other hand, if a stiffly accurate integrator is chosen, one has to solve nonlinear equations that are difficult to handle with, especially in connection with spectral methods [8]. In order to avoid these difficulties, the linearly implicit Runge-Kutta methods proposed in [6] combine an L-stable singly diagonally implicit Runge-Kutta method to treat the linear part with and explicit Runge-Kutta scheme for the nonlinear terms, in such a way that the overall method has good stability properties for spectral discretizations of advection-reaction-diffusion equations (see [6] for more details). These methods have Butcher tableaux

$$\begin{array}{c|c|c} \mathbf{c} & \mathcal{A} & \hat{\mathcal{A}} \\ \hline & \mathbf{b}^T & \mathbf{b}^T \end{array} \tag{2.3}$$

with $\mathcal{A} = (a_{ij})_{i,j=1}^{s+1}$, $\hat{\mathcal{A}} = (\hat{a}_{ij})_{i,j=1}$, $\mathbf{c} = [0, c_2, \ldots, c_{s+1}]^T$ and $\mathbf{b}^T = [0, b_2, \ldots, b_{s+1}]$. Matrix $\mathcal{A}$ is lower triangular with constant diagonal elements and has null first column, while $\hat{\mathcal{A}}$ is strictly lower triangular. The equations to advance a step of length k in the integration of (2.2) from t_n to $t_{n+1} = t_n + k$ take the form

$$\begin{aligned} \mathbf{Y}_1 &= \mathbf{y}_n, \\ \mathbf{Y}_i &= \mathbf{y}_n + k\left(\sum_{j=2}^{i} a_{ij} L\mathbf{Y}_j + \sum_{j=1}^{i-1} \hat{a}_{i,j}\mathbf{N}(t_n + kc_j, \mathbf{Y}_j)\right), \quad 2 \le i \le s+1, \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + k\left(\sum_{i=2}^{s+1} b_i L\mathbf{Y}_i + \sum_{i=2}^{s+1} b_i \mathbf{N}(t_n + kc_i, \mathbf{Y}_i)\right), \end{aligned}$$

where $\mathbf{Y}_i$ denote the internal stages. In our numerical experiments we use the linearly implicit Runge-Kutta method with classical order $p = 3$, stage order $q = 1$ and $s + 1 = 4$ stages proposed in [6]. Due to the special structure of matrices $\mathcal{A}$ and $\hat{\mathcal{A}}$, the method requires 4 evaluations of the nonlinear term per step, 3 matrix-vector multiplications per step and 3 linear systems with the same matrix to be solved at each step.

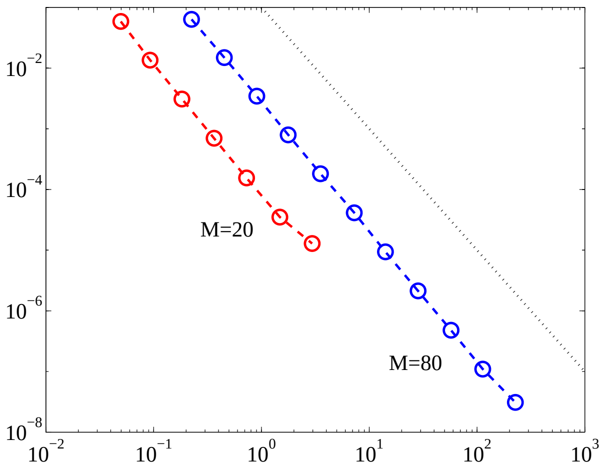

FIGURE 1. Error at $T = 10$ against CPU time for Burgers' equation

In order to observe the errors due to the time integrator, we fix the number of degrees of freedom for the spatial discretization and we consider time steps $k = 0.4, 0.2, \ldots$. In Figure 1 we have plotted error at $T = 10$ (measured in the L^2 norm) against CPU time for two different values of the number of degrees of freedom $M = 80$ and $M = 20$. Only data for which the error due to the spatial discretization is smaller than the error of the time integration have been plotted. We observe that the time integrator behaves as a second order method as corresponds to the slope -2 of both lines. Then, the order reduction is present because the classical order of the method is 3. We want to stress that although the stage order of the method is only 1, the errors shown in Figure 1 exhibit second order. This is due to the homogeneous boundary conditions satisfied by the exact solution of (2.1), which imply that the first correction (1.3) with $r = 1$ is already done (see [5], [13] for details).

3. How to avoid the order reduction in the nonlinear case

In this section we show how the strategy of [7] can be adapted to deal with certain families of nonlinear problems, including Burgers' equation.

Let us now consider an abstract initial boundary value problem of the form

$$(3.1) \qquad \begin{cases} u'(t) = \widetilde{A}u(t) + R(u(t)) + f(t), & 0 \le t \le T, \\ u(0) = u_0, & \\ \partial u(t) = g(t), & 0 \le t \le T, \end{cases}$$

with $\widetilde{A}$ and ∂ as in (1.1) and $R : D(R) \subset X \to X$ a nonlinear operator satisfying $D(\widetilde{A}) \subset D(R)$. For instance, Burgers' equation can be written in this abstract format as follows: $X = L^2([-1,1])$, $Y = \mathbb{R} \times \mathbb{R}$, $D(\widetilde{A}) = W^{2,2}([-1,1]) = \{ \varphi \in X \,/\, \varphi'' \in X \}$, $\widetilde{A}\varphi = \varphi''$, $R(\varphi) = \varphi\varphi'$ and $\partial\varphi = [\varphi(-1), \varphi(1)]^T$, $\varphi \in D(\widetilde{A})$. In this example $D(A) = W^{2,2}([-1,1]) \cap W_0^{1,2}([-1,1]) = \{ \varphi \in W^{2,2}([-1,1]) \,/\, \varphi(-1) = \varphi(1) = 0 \}$ and for $\xi = [\xi_0, \xi_1]^T \in Y$, $E\xi$ is the mapping sending $x \in [-1,1]$ to $\xi_0 + ((\xi_1 - \xi_0)/2)(x+1)$. Notice that in practice, R collects spatial lower order derivatives and nonlinear terms.

According to (1.3), if $u : [0,T] \to X$ is a solution of (3.1), let us define

$$\begin{aligned} v_2(t) &= E\partial u(t) + A^{-1}E\partial\widetilde{A}u(t) \\ &= E\partial u(t) + A^{-1}E\partial[u'(t) - R(u(t)) - f(t)]. \end{aligned}$$

Then $u^{*,2}(t) = u(t) - v_2(t)$ is the solution of the initial value problem

$$\left\{ \begin{array}{l} (u^{*,2})'(t) = Au^{*,2}(t) + f^{*,2}(t), \quad 0 \le t \le T, \\ u^{*,2}(0) = u_0 + A^{-1}E\partial f(0), \end{array} \right. \tag{3.2}$$

for an appropriate $f^{*,2}(t)$.

Let us now assume that $\partial u = 0$ and $\partial R(u) = 0$. This includes initial boundary value problems subject to either Dirichlet or Neumann homogeneous boundary conditions and with nonlinearities including reaction terms of the form $R(u)$ with R a smooth real function satisfying $R(0) = 0$, or other nonlinear terms such as $R(u) = (u \cdot \nabla)u$ appearing in Navier-Stokes equations or Burgers' type equations. Under these assumptions,

$$v_2(t) = -A^{-1}E\partial f(t)$$

and

$$f^{*,2}(t) = R(u(t)) + f(t) - \overbrace{\left[E\partial f(t) - A^{-1}E\partial f'(t)\right]}^{H(t)}. \tag{3.3}$$

Notice that $v_2(t)$ and $H(t)$ are computable in terms of the data but the nonlinear term depends on the solution $u(t)$ of the original problem (3.1). This is the main difference with respect to the linear case, where $f^{*,2}(t)$ was computable in terms of f and g and their derivatives (see (1.6)).

From the theoretical point of view this is not a serious drawback since $u(t) = u^{*,2}(t) + v_2(t)$ and $v_2(t)$ is computable in terms of $f(t)$. However, the computation of $v_2(t)$ during the time integration requires the solution of several additional stationary problems per step, which can be expensive in practice. In Section 4 we propose an alternative way of providing approximations to u and $u^{*,2}$ by time integrating (3.1) and (3.2) simultaneously using a linearly implicit Runge-Kutta method as those proposed in [6].

4. An efficient implementation of this strategy

For the time integration of (3.2), let us consider a linearly implicit Runge-Kutta method with Butcher tableaux (2.3). The equations to time integrate (3.2) from $t = 0$ to $t_N = Nk$ take the form:

For $n = 0, 1, 2, \ldots, N-1$,

$$\begin{aligned}
U_1^{*,2} &= u_n^{*,2},\\
U_i^{*,2} &= u_n^{*,2} + k\sum_{j=2}^{i} a_{ij} A U_j^{*,2}\\
&+ k\sum_{j=1}^{i-1} \hat{a}_{ij}\left[R(u(t_n+c_jk)) + f(t_n+c_jk) - H(t_n+c_jk)\right], \quad 2 \le i \le s+1,\\
u_{n+1}^{*,2} &= u_n^{*,2} + k\sum_{i=2}^{s+1} b_i \left[A U_i^{*,2} + R(u(t_n+c_ik)) + f(t_n+c_ik) - H(t_n+c_ik)\right],\\
t_{n+1} &= t_n + k,
\end{aligned}$$

with $t_0 = 0$ and $u_0^{*,2} = u_0 - v_2(0)$. Then, the approximation to the solution of (3.1) at t_N is given by

$$\hat{u}_N = u_N^{*,2} - A^{-1} E \partial f(t_N). \tag{4.1}$$

However, the nonlinear terms are not known because they depend on the exact solution of (3.1). What we propose in order to overcome this difficulty is to time integrate (3.1) and (3.2) simultaneously. More precisely,
For $n = 0, 1, 2, \ldots, N-1$,

$$\begin{aligned}
U_1 &= u_n,\\
U_1^* &= u_n^*,\\
U_i &= u_n + k\sum_{j=2}^{i} a_{ij} \widetilde{A} U_j\\
&+ k\sum_{j=1}^{i-1} \hat{a}_{ij}\left[R(U_j) + f(t_n+c_jk)\right], \quad 2 \le i \le s+1,\\
U_i^* &= u_n^* + k\sum_{j=2}^{i} a_{ij} A U_j^*\\
&+ k\sum_{j=1}^{i-1} \hat{a}_{ij}\left[R(U_j)) + f(t_n+c_jk) - H(t_n+c_jk)\right], \quad 2 \le i \le s+1,\\
u_{n+1} &= u_n + k\sum_{i=2}^{s+1} b_i \left[\widetilde{A} U_i + R(U_i) + f(t_n+c_ik)\right],\\
u_{n+1}^* &= u_n^* + k\sum_{i=2}^{s+1} b_i \left[A U_i^* + R(U_i) + f(t_n+c_ik) - H(t_n+c_ik)\right],\\
t_{n+1} &= t_n + k,
\end{aligned}$$

with $t_0 = 0$ and $u_0^* = u_0 - v_2(0)$. Now, the approximation to the solution of (3.1) at t_N is given by

$$\hat{\hat{u}}_N = u_N^* - A^{-1} E \partial f(t_N). \tag{4.2}$$

In this way, for $1 \leq j \leq s+1$, the term $R(u(t_n + c_j k))$ is approximated by $R(U_j)$, being U_j, $1 \leq j \leq s+1$, the internal stages of the linearly implicit Runge-Kutta method when applied to the original problem (3.1). Notice that the additional computational cost of this implementation, compared with the standard implementation of the linearly implicit method applied to (3.1), is just the computational cost of the solution of 3 linear systems per step, but with the same matrix used in the standard procedure. It is also necessary to evaluate the function $H(t)$ at three different time levels per step, but this can be efficiently done in parallel at the beginning of each step. Furthermore, although each evaluation of $H(t)$ requires the solution of 3 stationary problems, in the one dimensional case the computational cost of this task is almost negligible since (1.2) can be solved exactly.

On the other hand, in the implementation above, the term $R(u(t_n + c_j k))$ is approximated by $R(U_j)$, $1 \leq j \leq s+1$, which is only a second order approximation since the stage order of the linearly implicit Runge-Kutta method is 1. However, the following result holds:

Theorem 4.1. *Let $\hat{u}_N$ and $\hat{\hat{u}}_N$ be the numerical approximations to $u(t_N)$ defined by (4.1) and (4.2) respectively. If $\|\hat{u}_N - u(t_N)\| = O(k^3)$, then $\|\hat{\hat{u}}_N - u(t_N)\| = O(k^3)$.*

Proof. We will prove that $\|\hat{u}_N - \hat{\hat{u}}_N\| = O(k^3)$. First we observe that substracting (4.2) from (4.1) it is enough to prove that $\|u_N^{*,2} - u_N^*\| = O(k^3)$. To this end, we set $F^{*,2}(t_n) = \left\{f^{*,2}(t_n + c_i k)\right\}_{i=1}^{s+1}$, where $f^{*,2}(t)$ is defined by (3.3) and $F^*(t_n) = \{R(U_i) + f(t_n + c_i k) - H(t_n + c_i k)\}_{i=1}^{s+1}$, where $U_1, \ldots, U_{s+1}$ are the internal stages of the linearly implicit Runge-Kutta method when applied to (3.1) to advance from t_n to t_{n+1}. We also set $U^{*,2} = \{U_i^{*,2}\}_{i=1}^{s+1}$ and $U^* = \{U_i^*\}_{i=1}^{s+1}$. According to (3.3), $F^{*,2}(t_n) - F^*(t_n) = \{R(u(t_n + c_i k)) - R(U_i)\}_{i=1}^{s+1}$, being $u(t)$ the exact solution of (3.1). Since the stage order of the linearly implicit Runge-Kutta method is 1,

$$u(t_n + c_i k) - U_i = k^2 a_i(t_n) + O(k^3)$$

and then if R is smooth enough

$$F^{*,2}(t_n) - F^*(t_n) = k^2 \mathbf{a}^*(t_n) + O(k^3). \tag{4.3}$$

The vectors collecting the internal stages $U^{*,2}$ and U^* satisfy

$$U^{*,2} = (\mathbf{e} \otimes I) u_n^{*,2} + k(\mathcal{A} \otimes A) U^{*,2} + k(\hat{\mathcal{A}} \otimes I) F^{*,2}(t_n) \tag{4.4}$$

and

$$U^* = (\mathbf{e} \otimes I) u_n^* + k(\mathcal{A} \otimes A) U^* + k(\hat{\mathcal{A}} \otimes I) F^*(t_n), \tag{4.5}$$

respectively. Here $\mathcal{A}$ and $\hat{\mathcal{A}}$ denote the matrices of the Runge-Kutta tableaux in (2.3), $\mathbf{e} = [1, \ldots, 1]^T \in \mathbb{R}^{s+1}$ and I stands for the identity matrix. On the other hand, the approximations to the solution at t_{n+1} satisfy

$$u_{n+1}^{*,2} = u_n^{*,2} + k(\mathbf{b}^T \otimes A)U^{*,2} + k(\mathbf{b}^T \otimes I)F^{*,2}(t_n) \tag{4.6}$$

and

$$u_{n+1}^{*} = u_n^{*} + k(\mathbf{b}^T \otimes A)U^{*} + k(\mathbf{b}^T \otimes I)F^{*}(t_n), \tag{4.7}$$

respectively, where $\mathbf{b}^T$ are the weights of the linearly implicit Runge-Kutta method (2.3). Substracting (4.5) from (4.4) and using (4.3) we get

$$U^{*,2} - U^{*} = (\mathbf{e}\otimes I)[u_n^{*,2} - u_n^{*}] + k(\mathcal{A}\otimes A)[U^{*,2} - U^{*}] + k(\hat{\mathcal{A}}\otimes I)[k^2\mathbf{a}^*(t_n) + O(k^3)],$$

which leads to

$$\begin{aligned} U^{*,2} - U^{*} &= (I - k(\mathcal{A}\otimes A))^{-1}(\mathbf{e}\otimes I)[u_n^{*,2} - u_n^{*}] \\ &+ k^3(I - k(\mathcal{A}\otimes A))^{-1}(\hat{\mathcal{A}}\otimes I)\mathbf{a}^*(t_n) + O(k^4). \end{aligned} \tag{4.8}$$

On the other hand, substracting (4.7) from (4.6) and using again (4.3) we get

$$u_{n+1}^{*,2} - u_{n+1}^{*} = [u_n^{*,2} - u_n^{*}] + k(\mathbf{b}^T \otimes A)[U^{*,2} - U^{*}] + k(\mathbf{b}^T \otimes I)[k^2\mathbf{a}^*(t_n) + O(k^3)],$$

which after using (4.8) and some manipulations leads to

$$\begin{aligned} u_{n+1}^{*,2} - u_{n+1}^{*} &= \left(I + k\left(\mathbf{b}^T \otimes A\right)(I - k(\mathcal{A}\otimes A))^{-1}(\mathbf{e}\otimes I)\right)[u_n^{*,2} - u_n^{*}] \\ &+ k^3(\mathbf{b}^T \otimes I)\mathbf{a}^*(t_n) + O(k^4). \end{aligned}$$

Now, the variation of constants formula applied to the previous equality shows that

$$\begin{aligned} u_{n+1}^{*,2} - u_{n+1}^{*} &= (r(kA))^{n+1}[u_0^{*,2} - u_0^{*}] \\ &+ k^3\sum_{j=0}^{n}(r(kA))^{n-j}[(\mathbf{b}^T \otimes I)\mathbf{a}^*(t_j) + O(k)], \end{aligned}$$

where $r(kA) = I + k\left(\mathbf{b}^T \otimes A\right)(I - k(\mathcal{A}\otimes A))^{-1}(\mathbf{e}\otimes I)$. Notice that $r(z)$ is nothing but the stability function of the implicit part of the linearly implicit Runge-Kutta method (2.3). As long as $u_0^{*,2} = u_0^{*}$, and using summation by parts, we get

$$\begin{aligned} u_{n+1}^{*,2} - u_{n+1}^{*} &= k^3 R_n(kA)(\mathbf{b}^T \otimes I)\mathbf{a}^*(0) \\ &+ k^3\sum_{j=1}^{n} R_{n-j}(kA)(\mathbf{b}^T \otimes I)[\mathbf{a}^*(t_j) - \mathbf{a}^*(t_{j-1})] \\ &+ O(k^3), \end{aligned}$$

with $R_j(kA) = \sum_{l=0}^{j}(r(kA))^l$ for $0 \le j \le n$. Since $[\mathbf{a}^*(t_j) - \mathbf{a}^*(t_{j-1})] = O(k)$, it is clear that for $0 \le n \le N-1$,

$$\|u_{n+1}^{*,2} - u_{n+1}^{*}\| = O(k^3),$$

as we wanted to prove. □

5. Numerical results

In order to illustrate the performance of the strategy described in the previous section, we consider again the forced Burgers' equation (2.1) with the same initial data and forcing term as in Section 2. We repeat the experiment of Section 2 but with the new implementation of the time integrator. In Figure 2 we have plotted error at $T = 10$ (measured in the L^2 norm) against CPU time for $M = 80$ and $M = 20$ degrees of freedom and step-sizes $k = 0.4, 0.2, \ldots, 0.003125$ for $M = 80$ and $k = 0.4, 0.2, \ldots, 0.025$ for $M = 20$. Only data for which the error due to the spatial discretization is smaller than the error caused by the time integrator have been plotted. We observe that the time integrator now behaves as a third order method as corresponds to the slope -3 of both lines. Then, the order reduction has been avoided.

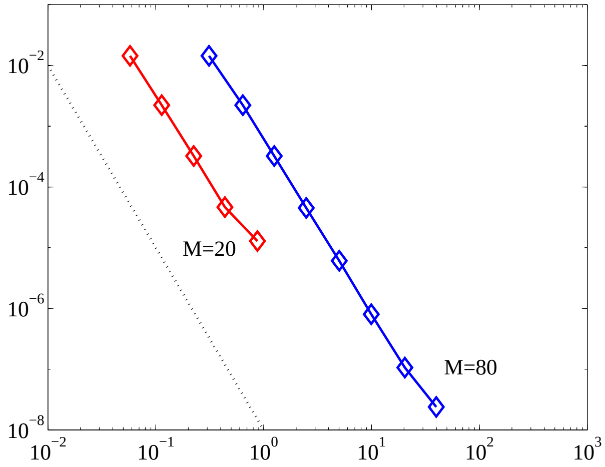

FIGURE 2. Error at $T = 10$ against CPU when integrating (3.2) for Burgers' equation

In Figure 3 we have plotted the results obtained using both procedures, in order to see which of them is the most efficient. We use diamonds joined by a solid line for the data obtained with the implementation without order reduction and circles joined by a dashed line for the data generated with the linearly implicit method applied to Burgers' equation. We see that independently of the number of degrees of freedom used in the spatial discretization, the new implementation of the time integrator is more efficient than the standard integration of the original problem. We observe not only the third order of the new implementation, as expected, but we also observe that the cheapest way to get errors under a prescribed tolerance always corresponds to the time integration using the new procedure. We conclude that the strategy proposed in [7] to avoid the order reduction of

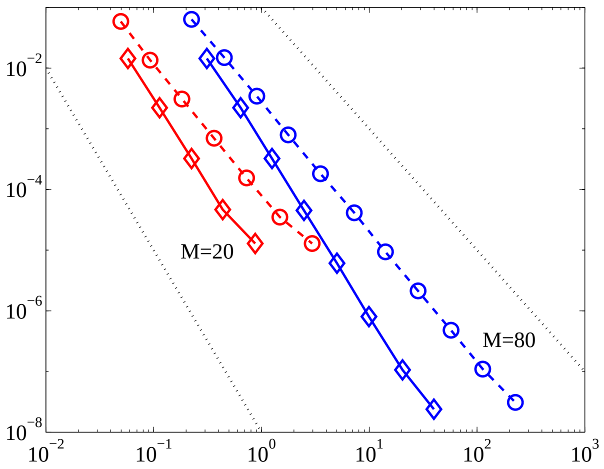

FIGURE 3. Efficiency diagram comparing both procedures

Runge-Kutta methods and adapted in the present paper to also deal with nonlinear problems is interesting not only from a theoretical point of view but also as an efficient procedure to time integrate one dimensional nonlinear initial boundary value problems.

References

[1] S. Abarbanel, D. Gottlieb & M. H. Carpenter, *On the removal of boundary errors caused by Runge-Kutta integration of nonlinear partial differential equations*, SIAM J. Sci. Comput. **17** (1996), 777–782.

[2] I. Alonso-Mallo, *Explicit single step methods with optimal order of convergence for partial differential equations*, Appl. Numer. Math. **31** (1999), 117–131.

[3] I. Alonso-Mallo, *Rational methods with optimal order of convergence for partial differential equations*, Appl. Numer. Math. **35** (2000), 265–292.

[4] I. Alonso-Mallo & B. Cano, *Spectral/Rosenbrock discretizations without order reduction for linear parabolic problems*, Appl. Numer. Math. **41** (2002), 247–268.

[5] I. Alonso-Mallo & C. Palencia, *Optimal orders of convergence for Runge-Kutta methods and linear, initial boundary value problems*, Appl. Numer. Math. (to appear).

[6] M. P. Calvo, J. de Frutos & J. Novo, *Linearly implicit Runge-Kutta methods for advection-reaction-diffusion equations*, Appl. Numer. Math. **37** (2001), 539–549.

[7] M. P. Calvo & C. Palencia, *Avoiding the order reduction of Runge-Kutta methods for linear initial boundary value problems*, Math. Comput. **71** (2002), 1529–1543.

[8] C. Canuto, M. Hussaini, A. Quarteroni & T. Zang, *Spectral methods in fluid dynamics*, Springer-Verlag, 1987.

[9] M. H. Carpenter, D. Gottlieb, S. Abarbanel & W. S. Don, *The theoretical accuracy of Runge-Kutta time discretizations for the initial boundary value problem: a study of the boundary error*, SIAM J. Sci. Comput. **16** (1995), 1241–1252.

[10] E. Hairer & G. Wanner, *Solving Ordinary Differential Equations II, Stiff and Differential-Algebraic Problems*, 2nd Edition, Springer, 1996.

[11] S. L. Keeling, *Galerkin/Runge-Kutta discretizations for parabolic equations with time-dependent coefficients*, Math. Comput. **52** (1989), 561–586.

[12] S. L. Keeling, *Galerkin/Runge-Kutta discretizations for semilinear parabolic equations*, SIAM J. Numer. Anal. **27** (1990), 394–418.

[13] A. Ostermann & M. Roche, *Runge-Kutta methods for partial differential equations and fractional orders of convergence*, Math. Comput. **64** (1995), 591–599.

[14] C. Palencia & I. Alonso-Mallo, *Abstract initial boundary value problems*, Proc. Royal Soc. Edinburgh **124A** (1994), 879–908.

[15] D. Pathria, *The correct formulation of intermediate boundary conditions for Runge-Kutta time integration of initial boundary value problems*, SIAM J. Sci. Comput. **18** (1997), 1255–1266.

[16] J. M. Sanz-Serna, J. G. Verwer & W. H. Hundsdorfer, *Convergence and order reduction of Runge-Kutta schemes applied to evolutionary problems in partial differential equations*, Numer. Math. **50** (1986), 405–418.

[17] J. Shen, *Efficient spectral-Galerkin method II. Direct solvers of second and four order equations using Chebyshev polynomials*, SIAM J. Sci. Comput. **16** (1995), 74–87.

[18] J. G. Verwer & J. Lang, *ROS3P – An accurate third-order Rosenbrock solver designed for parabolic problems*, BIT **41** (2001), 731–738.

Acknowledgment

The authors are grateful to Prof. C. Palencia for helpful discussions and suggestions.

Departamento de Matemática Aplicada y Computación, Universidad de Valladolid, Valladolid, Spain
E-mail address: maripaz@mac.cie.uva.es

Departamento de Matemática Aplicada Fundamental, Universidad de Valladolid, Valladolid, Spain
E-mail address: frutos@mac.cie.uva.es

Departamento de Matemáticas, Universidad Autónoma de Madrid, Madrid, Spain
E-mail address: julia.novo@uam.es