Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

MAS

Modelling, Analysis and Simulation

Modelling, Analysis and Simulation

Time staggering for wave equations revisited

J.G. Verwer

REPORT MAS-E0710 JUNE 2007

Centrum voor Wiskunde en Informatica (CWI) is the national research institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organisation for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3703

Time staggering for wave equations revisited

Abstract:
Staggering in numerical methods for wave equations generally enhances accuracy and stability. This note is about time staggering. We assess a fourth-order, explicit, time-staggered method, while focussing on a class of second-order wave equations. Alternative explicit integration methods for this class belong to the Runge-Kutta-Nyström (RKN) family and we have selected three explicit RKN methods for our assessment of the time-staggered method. Compared to these three explicit RKN methods, the time-staggered method possesses a substantially longer stability interval. Our aim is to examine whether this advantage can be expected to borne out in actual computation.

2000 Mathematics Subject Classification: 65L06, 64M06, 65M12, 65M20
1998 ACM Computing Classification System: G.1.7, G.1.8
Keywords and Phrases: Second-order wave equations, explicit time integration, time staggering
Note: Work carried out within theme MAS1

Time Staggering for Wave Equations Revisited *

J.G. Verwer†
Centrum voor Wiskunde en Informatica (CWI)
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

June 22, 2007

Abstract

Staggering in numerical methods for wave equations generally enhances accuracy and stability. This note is about time staggering. We assess a fourth-order, explicit, time-staggered method, while focussing on a class of second-order wave equations. Alternative explicit integration methods for this class belong to the Runge-Kutta-Nyström (RKN) family and we have selected three explicit RKN methods for our assessment of the time-staggered method. Compared to these three explicit RKN methods, the time-staggered method possesses a substantially longer stability interval. Our aim is to examine whether this advantage can be expected to borne out in actual computation.

2000 Mathematics Subject Classification: 65L06, 64M06, 65M12, 65M20.
1998 ACM Computing Classification System: G.1.7, G.1.8.

Keywords and Phrases: Second-order wave equations, explicit time integration, time staggering
Note: Work carried out within theme MAS1.

1 Introduction

This paper is a sequel to [12]. In that paper we examined a fourth-order, explicit, time-staggered integration method proposed in [6, 7] for integrating special partitioned ODE systems

$$\begin{aligned} u' &= f(t,v)\,, \\ v' &= g(t,u)\,, \end{aligned} \tag{1.1}$$

in particular systems representing semi-discrete wave equations. For systems of first-order wave equations, this time-staggered method was compared with the classical Runge-Kutta method and with a fourth-order, explicit, symmetric-composition method based on symplectic Euler. The time-staggered and symmetric-composition method turned out to perform notably better than the classical Runge-Kutta method, whereas no real distinction was observed in their mutual performance. Prior to the investigation in [12], our expectation was a notable difference in favor of the time-staggered method, as it has a substantially

*Unpublished note

†Jan.Verwer@cwi.nl, http://homepages.cwi.nl/∼janv

larger scaled stability interval and also suffers less from order reduction in the presence of time-dependent boundary conditions.

The aim of the current paper, therefore, is to shed more light on the potential advantage of time staggering. Herewith we will focus on semi-discrete, second-order wave equation systems $u'' = g(t, u)$, written in the first-order form

$$\begin{aligned} u' &= v\,, \\ v' &= g(t, u)\,. \end{aligned} \tag{1.2}$$

We will present a brief numerical comparison of the time-staggered method to three existing, explicit Runge-Kutta-Nyström (RKN) methods. Two of these were especially designed for systems of type (1.2), while the third is the symmetric-composition method from [12], which for system (1.2) belongs to the RKN family.

2 The integration methods

In this section we give the formulations for system (1.2) of the explicit staggered method and of three explicit RKN methods, including the one from [12] based on symmetric composition of symplectic Euler.

2.1 StaggeredLF4

Time staggering means approximating u and v at interlaced time levels, one after the other. Following [6, 7, 12] we choose integer levels t_n for u and half-integer levels $t_{n+1/2}$ for v for $n = 0, 1, \ldots$. Level t_n denotes time $t_n = n\tau$ with constant step size τ. Let u_n and $v_{n+1/2}$ be the approximation to $u(t_n)$ and $v(t_{n+1/2})$, respectively. A well-known time-staggered integration method for system (1.2) is the second-order, explicit, staggered leap-frog rule

$$\begin{aligned} u_{n+1} &= u_n + \tau v_{n+1/2}\,, \\ v_{n+3/2} &= v_{n+1/2} + \tau g(t_{n+1}, u_{n+1})\,. \end{aligned} \tag{2.1}$$

This method thus steps from $(u_n, v_{n+1/2})$ to $(u_{n+1}, v_{n+3/2})$ with step size τ. Within geometric integration it is known as the Störmer-Verlet scheme [8]. The missing value $v_{1/2}$ at the start is to be provided by an appropriate one-step method.

The fourth-order staggered method from [6, 7] follows the same recipe, except that it uses internal stages. In [12] we gave two representations. The one which best reveals its relation with the staggered leap-frog scheme reads

$$\begin{aligned} U_1 &= u_n + \tau v_{n+1/2} & V_1 &= v_{n+1/2} + \tau g(t_{n+1}, u_{n+1}) \\ V_2 &= v_{n+1/2} - \tau g(t_n, u_n) & U_2 &= u_{n+1} - \tau v_{n+1/2} \\ U_3 &= u_n + \tau V_2 & V_3 &= v_{n+1/2} + \tau g(t_n, U_2) \\ V_4 &= v_{n+1/2} + \tau g(t_{n+1}, U_1) & U_4 &= u_{n+1} + \tau V_1 \\ U_5 &= u_n + \tau V_4 & V_5 &= v_{n+1/2} + \tau g(t_{n+2}, U_4) \\ u_{n+1} &= \tfrac{22}{24}U_1 + \tfrac{1}{24}U_3 + \tfrac{1}{24}U_5 & v_{n+3/2} &= \tfrac{22}{24}V_1 + \tfrac{1}{24}V_3 + \tfrac{1}{24}V_5 \end{aligned} \tag{2.2}$$

In the formula at the left, defining u_{n+1}, odd and even numbered stages are used for u and v, respectively, while in the formula at the right, which defines $v_{n+3/2}$, it is the other way

around. The approximation $v_{n+3/2}$ uses the same coefficients as u_{n+1}, the only difference being that u and v are interchanged and all time levels used for $v_{n+3/2}$ are shifted forward with 1/2.

The method can be interpreted as a particular fourth-order correction to second-order staggered leap-frog. This follows by rewriting the formulas for u_{n+1} and $v_{n+3/2}$ as

$$\begin{aligned} u_{n+1} &= u_n + \tau v_{n+1/2} + \\ &\quad \tfrac{1}{24}\tau^2 \left(g(t_{n+1}, U_1) - g(t_n, u_n)\right) , \\ v_{n+3/2} &= v_{n+1/2} + \tau g(t_{n+1}, u_{n+1}) + \\ &\quad \tfrac{1}{24}\tau \left(g(t_n, U_2) - 2g(t_{n+1}, u_{n+1}) + g(t_{n+2}, U_4)\right) . \end{aligned} \tag{2.3}$$

Like for staggered leap-frog, the missing value $v_{1/2}$ at the start is to be provided by an appropriate one-step method. Note that although (2.2) is written in 5 stages, it can be implemented with 4 evaluations of g, because $g(t_{n+1}, u_{n+1})$ can be saved to provide V_2 for free at the next step. Also note that steps in the negative direction are taken. In the remainder we will refer to (2.2) as StaggeredLF4, similar as in [12].

2.2 Two explicit Runge-Kutta-Nyström methods

For the assessment of StaggeredLF4 we use two explicit RKN methods from the literature. Such methods read [8]

$$\begin{aligned} U_i &= u_n + \tau\gamma_i v_n + \tau^2 \textstyle\sum_{j=1}^{i-1} \alpha_{ij} g(t_n + \gamma_j\tau, U_j) , \qquad i = 1, \ldots, s , \\ u_{n+1} &= u_n + \tau v_n + \tau^2 \textstyle\sum_{i=1}^{s} \beta_i g(t_n + \gamma_i\tau, U_i) , \\ v_{n+1} &= v_n + \tau \textstyle\sum_{i=1}^{s} b_i g(t_n + \gamma_i\tau, U_i) , \end{aligned} \tag{2.4}$$

where s is the number of stages and

$$\begin{aligned} &\gamma_1 = 0 , \quad \gamma_s = 1 , \\ &\beta_j = \alpha_{sj} , \quad j = 1, \ldots, s-1 , \\ &\alpha_{ij} = b_j(\gamma_i - \gamma_j) , \quad i > j . \end{aligned} \tag{2.5}$$

The first and second condition give the FSAL (First Same As Last) property, so that with s stages only $s-1$ evaluations of g are needed. The third renders a method symplectic.

We have chosen a fourth-order, five-stage method from [3] and a fifth-order, seven-stage method from [4]. Both are believed to be sufficiently representative within the explicit RKN class for a numerical assessment of StaggeredLF4. In the remainder the fourth-order, five-stage method will be called RKN45. Its coefficients are, see [3],

$$\begin{array}{lllllll} \gamma_1 &=& 0 & b_1 &=& 0.061758858135626325 \\ \gamma_2 &=& 0.205177661542286386 & b_2 &=& 0.338978026553643355 \\ \gamma_3 &=& 0.608198943146500973 & b_3 &=& 0.614791307175577566 \\ \gamma_4 &=& 0.487278066807586965 & b_4 &=& -0.140548014659373380 \\ \gamma_5 &=& 1 & b_5 &=& 0.125019822794526133 \end{array} \tag{2.6}$$

The fifth-order, seven-stage method will be called RKN57. Its coefficients are, see [4],

$$\begin{array}{rclrcl}
\gamma_1 &=& 0 & b_1 &=& 0.062812135702683290 \\
\gamma_2 &=& 0.217962139017564600 & b_2 &=& 0.378898313125257500 \\
\gamma_3 &=& 0.442470370825524200 & b_3 &=& 0.275452851526134000 \\
\gamma_4 &=& 1.478460559438898000 & b_4 &=& -0.001585299574780513 \\
\gamma_5 &=& 0.340000000000000000 & b_5 &=& -0.178570403852761800 \\
\gamma_6 &=& 0.700000000000000000 & b_6 &=& 0.347999583419883100 \\
\gamma_7 &=& 1 & b_7 &=& 0.114992819653584400
\end{array} \tag{2.7}$$

2.3 A symmetric-composition RKN method

Our third method used to assess StaggeredLF4, is the 4th-order, symmetric-composition method constructed in [12]. It is based on symplectic Euler and uses a coefficient set

$$\begin{aligned}
&\beta_1 = \alpha_5 = \tfrac{14-\sqrt{19}}{108}\,, \quad \alpha_1 = \beta_5 = \tfrac{146+5\sqrt{19}}{540}\,, \\
&\beta_2 = \alpha_4 = \tfrac{-23-20\sqrt{19}}{270}\,, \quad \alpha_2 = \beta_4 = \tfrac{-2+10\sqrt{19}}{135}\,, \quad \beta_3 = \alpha_3 = \tfrac{1}{5}\,,
\end{aligned} \tag{2.8}$$

from [9], which is known to give particularly small error terms [8]. In [12] we have examined two alternative coefficient sets. These were discarded due to a smaller scaled stability boundary. Let $U_0 = u_n, V_0 = v_n$. Adjusted for system (1.2), the method from [12] reads

$$\left.\begin{array}{rcl}
V_k &=& V_{k-1} + \eta_k \tau g(t_n + c_k\tau, U_{k-1}) \\
U_k &=& U_{k-1} + \delta_k \tau V_k\,,
\end{array}\right\} \quad k = 1\,,\ldots,s\,, \\
\begin{array}{rcl}
u_{n+1} &=& U_s\,, \\
v_{n+1} &=& V_s + \eta_{s+1}\tau g(t_n + \tau, u_{n+1})\,,
\end{array} \tag{2.9}$$

where $s = 5$ and

$$\begin{aligned}
&\eta_k = \alpha_{k-1} + \beta_k\,, \ \ \delta_k = \alpha_k + \beta_k\,, \ \ c_k = \textstyle\sum_{j=1}^{k-1}\delta_j\,, \quad k = 1\,,\ldots,s\,, \\
&\alpha_0 = 0\,, \ \ \eta_{s+1} = \alpha_s\,, \ \ c_0 = 0\,.
\end{aligned} \tag{2.10}$$

Because $g(t_{n+1}, u_{n+1})$ can be saved and $c_0 = 0$, the method can be seen to require $s = 5$ evaluations of g per step. Following [12], in the remainder we will refer to (2.9) as SymmetricCO4. Note that by construction (2.9) is symmetric and symplectic, while it can also be written as a six-stage, explicit RKN method (2.4)-(2.5) with generating coefficients

$$b_i = \eta_i\,, \quad \gamma_i = \sum_{j=1}^{i-1}\delta_j\,, \qquad i = 1\,,\ldots,6\,. \tag{2.11}$$

3 Linear stability

For stability we consider the linear model system

$$\begin{aligned}
u' &= v\,, \\
v' &= Bu\,,
\end{aligned} \tag{3.1}$$

where B is a symmetric, negative-definite matrix approximating a second-order elliptic differential operator. Since by assumption B is normal we may write $B = XDX^{-1}$, with D the diagonal eigenvalue matrix and X unitary. This decomposition carries over to all numerical methods considered here, so that for stability analysis we may as well consider the 2×2 model problem

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \mu & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}, \tag{3.2}$$

where μ represents any eigenvalue of B. Stability results for (3.2) then translate directly into stability results in L_2 for system (3.1).

The 2×2 matrix is decomposed as

$$\begin{pmatrix} 0 & 1 \\ \mu & 0 \end{pmatrix} = \begin{pmatrix} \sqrt{\mu^{-1}} & -\sqrt{\mu^{-1}} \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \sqrt{\mu} & 0 \\ 0 & -\sqrt{\mu} \end{pmatrix} \begin{pmatrix} \sqrt{\mu^{-1}} & -\sqrt{\mu^{-1}} \\ 1 & 1 \end{pmatrix}^{-1}. \tag{3.3}$$

The exponential matrix of the diagonal matrix immediately shows that the real part of $\sqrt{\mu}$ must be zero for stability of (3.2) (no growing exponentials). Consequently, the eigenvalues μ must be real negative. With B symmetric, negative-definite, this condition is fulfilled. From this transformation we also derive the exact solution of (3.2) as

$$\begin{pmatrix} u(t_{n+1}) \\ v(t_{n+1}) \end{pmatrix} = \mathcal{M}_e \begin{pmatrix} u(t_n) \\ v(t_n) \end{pmatrix}, \quad \mathcal{M}_e = \begin{pmatrix} \cosh(\tau\sqrt{\mu}) & \sqrt{\mu^{-1}}\sinh(\tau\sqrt{\mu}) \\ \sqrt{\mu}\sinh(\tau\sqrt{\mu}) & \cosh(\tau\sqrt{\mu}) \end{pmatrix}. \tag{3.4}$$

For all considered methods, the numerical counterpart of (3.4) takes the form of a recurrence

$$\begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} = \mathcal{M} \begin{pmatrix} u_n \\ v_n \end{pmatrix}, \tag{3.5}$$

where the 2×2 matrix $\mathcal{M}$ determines the stability of the method under consideration. The method is stable if $\mathcal{M}$ is power bounded, which holds if and only if its two eigenvalues lie on the unit disc and differ from one another when they lie on the unit circle (the root condition).

Similar as for the exact solution operator, the amplification operator $\mathcal{M}$ for all four numerical methods is stable only if $\sqrt{\mu}$ is purely imaginary. This leads to an interval condition on the purely imaginary values $z = \tau\sqrt{\mu}$, which defines the (open) imaginary stability interval $(-i\beta_{Im}, i\beta_{Im})$ with stability boundary β_{Im}. For $z \in (-i\beta_{Im}, i\beta_{Im})$ the two eigenvalues lie on the unit circle and for $z = \pm i\beta_{Im}$ they coincide and thus violate the root condition.

The boundary β_{Im} of StaggeredLF4 is known in closed form, $\beta_{Im} = 16^{1/3} + 32^{1/3} \approx 5.69$ [12]. For its competitors we have determined an accurate approximation by a numerical search. Table 3.1 contains the results. What counts is the scaled stability boundary $\beta_{Im,s}$, which is β_{Im} divided by the number of g-evaluations per step. We see that StaggeredLF4 stands out in the sense that its scaled stability boundary is substantially larger than that of its three competitors. Note that these three appear to possess a nearly equal β_{Im}.

Remark 3.1 For the three RKN methods we observed a peculiar behavior of the maximum modulus of the two eigenvalues of $\mathcal{M}$. Figure 3.1 shows the maximum modulus plotted as a function of $z = |\tau\sqrt{\mu}|$ for an interval substantially larger than $[0, \beta_{Im}]$. We see that

	order	g-evals	β_{Im}	$\beta_{Im,s}$
StaggeredLF4	4	4	5.69	1.42
SymmetricCO4	4	5	3.00	0.60
RKN45	4	4	3.04	0.76
RKN57	5	6	3.03	0.50

Table 3.1: Stability boundaries.

for most of the interval the modulus is equal to one, except at the end point and in a small neighborhood of $z = \beta_{Im}$, where a maximum slightly greater than one is found. Apparently, an interval of $\tau\sqrt{\mu}$ - values of substantial length exists outside the stability interval $(-i\beta_{Im}, i\beta_{Im})$, for which the maximum modulus is equal to one. The maximum modulus growths to infinity at the end point of this interval, rather than directly beyond β_{Im}. ◇

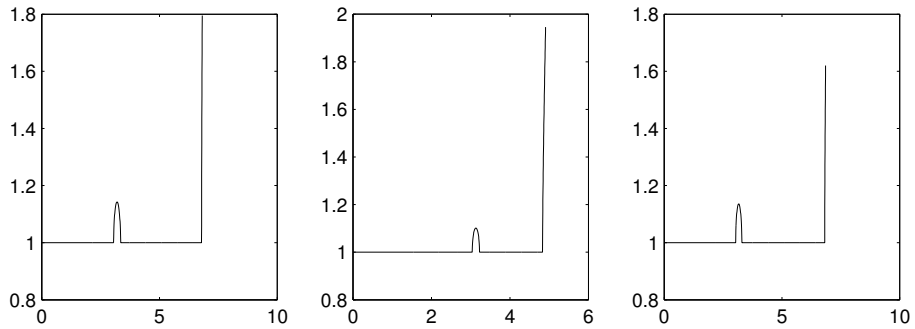

Figure 3.1: Maximum modulus of eigenvalues of $\mathcal{M}$ plotted as function of $z = |\tau\sqrt{\mu}|$. Left plot for RKN45, middle plot for RKN57, right plot for SymmetricCO4.

4 Numerical illustration

We have applied the four methods to the Sine-Gordon equation

$$\phi_{tt} = \phi_{xx} - \sin(\phi)\,, \qquad t > 0\,, \quad -L < x < L\,, \tag{4.1}$$

assuming Dirichlet boundary conditions and the so-called breather solution

$$\phi(x,t) = 4\tan^{-1}\Big(\frac{\sqrt{1-\omega^2}}{\omega}\,\frac{\cos(\omega t)}{\cosh(x\sqrt{1-\omega^2})}\Big)\,, \qquad \omega = 0.9\,, \tag{4.2}$$

which consists of an oscillating peak centered at $x = 0$ and vanishing outward at $x = \pm L$ for L sufficiently large. To illustrate the influence of time-dependent boundary values on the accuracy, we will compute the breather solution for $0 < t \leq 8\pi$ using $L = 10\pi$ and $L = \pi$. In the first case we do have (numerically) zero boundary values, while for $L = \pi$ the boundary values oscillate in time, see Figure 4.1. Hence, in the first case no order reduction effects will occur, whereas for $L = \pi$ this will happen.

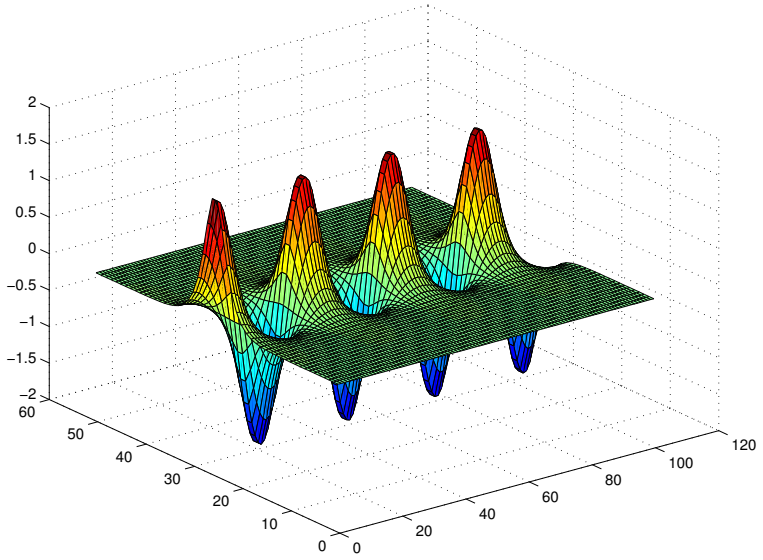

Figure 4.1: The breather solution of the Sine-Gordon equation for $0 \le t \le 8\pi\,, L = 10\pi$.

Let $h = 2L/(N+1)$, $x_i = -L + ih$ for $i = 0, 1, \ldots, N+1$, and let $u_i(t)$ and $v_i(t)$ denote the semi-discrete approximations to $\phi(x_i, t)$ and $\psi(x,t) = \phi_t(x_i, t)$, respectively. We discretize $\psi_t = \phi_{xx} - \sin(\phi)$ in space with the fourth-order implicit scheme

$$\begin{aligned} &\tfrac{1}{12}(v'_{i-1} + 10v'_i + v'_{i+1}) = \tfrac{1}{h^2}\left(u_{i-1} - 2u_i + u_{i+1}\right) - \\ &\tfrac{1}{12}\left(\sin(u_{i-1}) + 10\sin(u_i) + \sin(u_{i+1})\right)\,, \quad i = 1\,, \ldots\,, N\,. \end{aligned} \tag{4.3}$$

The boundary values v'_0, v'_{N+1} and u_0, u_{N+1} are described from the exact solution. Arranging the unknowns u_i, v_i in vectors u, v of length N, we then arrive at the system

$$\begin{aligned} u' &= v\,, \\ Mv' &= Au + Ms(u) + b(t)\,, \end{aligned} \tag{4.4}$$

with tridiagonal matrices M and A, $s(u) = (\sin(u_1)\,, \ldots\,, \sin(u_N))^T$, and $b(t)$ the vector containing boundary data. The spectral radius of $M^{-1}A$ equals $6/h^2$, approximately, resulting in the following maximal step sizes for stability

$$\tau = \begin{cases} 5.69h/\sqrt{6} \;\approx\; 2.32h & \text{StaggeredLF4} \\ 3.00h/\sqrt{6} \;\approx\; 1.22h & \text{SymmetricCO4} \\ 3.04h/\sqrt{6} \;\approx\; 1.24h & \text{RKN45} \\ 3.03h/\sqrt{6} \;\approx\; 1.24h & \text{RKN57} \end{cases} \tag{4.5}$$

With a minor adjustment to hit the chosen output time $t = 8\pi$ within an integer number of steps, the step sizes (4.5) are used in the numerical tests. SymmetricCO4 provides the start vector $v_{1/2}$ needed for StaggeredLF4. Because the spatial discretization is of order four, the two additional stages, which RKN57 uses for its order five compared to RKN45, seem

redundant. The results presented below contradict this. Apparently, the size of the error coefficients plays an important role, in particular when order reduction occurs.

Fig. 4.2 shows convergence results at time $t = 8\pi$ for component ϕ for $L = 10\pi$ (left plot) and $L = \pi$ (right plot). The plots are based on efficiency. That is, we plot maximum norm global errors (PDE solution minus fully discrete solution) versus computational work (number of integration steps times number of g-evaluations per step times number of spatial grid points). The marks in the plots correspond with $N = 40, 80, \ldots, 1280$ and associated values of the temporal step size τ according to (4.5). Note that for the two values of L we use the same numbers of grid points, resulting in ten times smaller values of h and τ in the right plot compared to the left one.

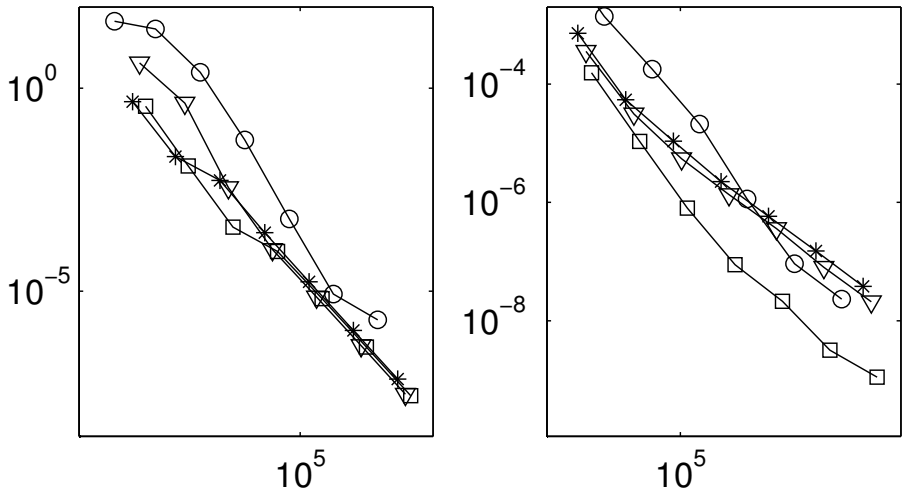

Figure 4.2: Loglog convergence plots for the Sine-Gordon problem. StaggeredLF4 o-marks, SymmetricCO4 ∇-marks, RKN45 $*$-marks, RKN57 $\square$-marks.

Let us first consider the left plot, where all four methods should reveal the fourth-order convergence for $\tau, h \to 0$. This indeed happens for RKN45, RKN57 and SymmetricCO4, while in the asymptotic regime the fourth-order spatial error dominates. StaggeredLF4 shows a somewhat irregular convergence behavior and is the least efficient. Its inaccuracy on the coarser grid is partly due to the larger step sizes used, according to (4.5).

As observed above, the results of the right plot are computed with tenfold smaller values for h and τ, resulting in higher accuracies. The expected order reduction takes place.[1] For $\tau, h \to 0$ the temporal order of RKN45, RKN57 and SymmetricCO4 goes down to two. It takes a while before this reduction becomes visible, especially for RKN57. This is due to particularly small error coefficients multiplying the elementary differentials that cause the reduction. It is obvious that the 5th-order method RKN57 is most efficient. This was not expected, due its higher workload per step and its notably smaller scaled stability interval, especially compared to StaggeredLF4, see Table 3.1. RKN57 clearly benefits from its higher order and from having very small error coefficients.

[1] In simple cases like our current test problem, the order reduction can be repaired by transforming the problem to one with a vanishing solution at the boundary [11]. See also [1, 10, 2] for more involved boundary correction approaches.

Also StaggeredLF4 suffers from reduction. We conjecture that in the limit $\tau, h \to 0$ the order goes down to three, which is the same as for first-order systems. A proof of this conjecture should be obtainable with the error analysis material given in [12]. It is obvious, however, that in the current test StaggeredLF4 does not compete with RKN57. Although StaggeredLF4 does beat SymmetricCO4 and RKN45 in the right plot for $\tau, h \to 0$, its overall performance, and hence the higher-order time staggering, works out less than we had hoped for.

5 Conclusion

Our aim has been to assess StaggeredLF4 for semi-discrete wave equations of the special type (1.2). Unfortunately, numerical results indicate that in general one cannot expect the method to outperform well-designed, classical Runge-Kutta-Nyström (RKN) methods from the ODE field, even though it has a larger scaled-stability interval and suffers less from order reduction, at least in the limit to convergence. Albeit limited, this experience is in line with what we observed in [12].

Of interest is the convergence behavior for 'stiff' source terms, in particular that of StaggeredLF4. To illustrate this once more, we conclude the paper with a numerical example for a purely contrived, academic toy problem. Consider, for $t > 0$, the initial-value problem

$$\begin{aligned} u' &= v\,, \\ v' &= g''(t) + \mu(u - g(t))\,, \end{aligned} \tag{5.1}$$

where g is any given, smooth function. By assigning the initial conditions $u(0) = g(0), v(0) = g'(0)$, we have as solution $u(t) = g(t), v(t) = g'(t)$. This problem is akin to the Prothero-Robinson test problem from the stiff ODE field, which was proposed to study order reduction of Runge-Kutta methods, see [5].

Applying all four methods tested earlier, we have integrated (5.1) for the choice $g(t) = e^t$ over the interval $0 \le t \le 2$, using

$$\begin{aligned} \tau &= 2^{-k}\,, \\ \mu &= -9\tau^{2k}\,, \end{aligned} \qquad k = 1, 2, \ldots, 12\,.$$

Hence, to illustrate the order reduction, we let

$$\tau \to 0\,, \quad |\mu| \to \infty \qquad \text{such that} \qquad \tau^2\mu = \text{constant}\,. \tag{5.2}$$

Figure 5.1 shows absolute errors for u at $t = 2$, plotted against computational work (number of time steps times number of evaluations of g per step). In this very special, academic case, StaggeredLF4 clearly comes out as best. We owe this to the fact that this method is less sensitive to order reduction than the RKN methods. Its convergence behavior is irregular, however, compared to that of the RKN methods, which converge roughly with order two. The irregular behavior of StaggeredLF4 may be due to cancellation effects. For understanding the convergence behavior under the asymptotics (5.2), the approach used in [12] can be used. Finally, the results obviously depend on the choice of g, that is, with another choice of g the performance of StaggeredLF4 may again be less.

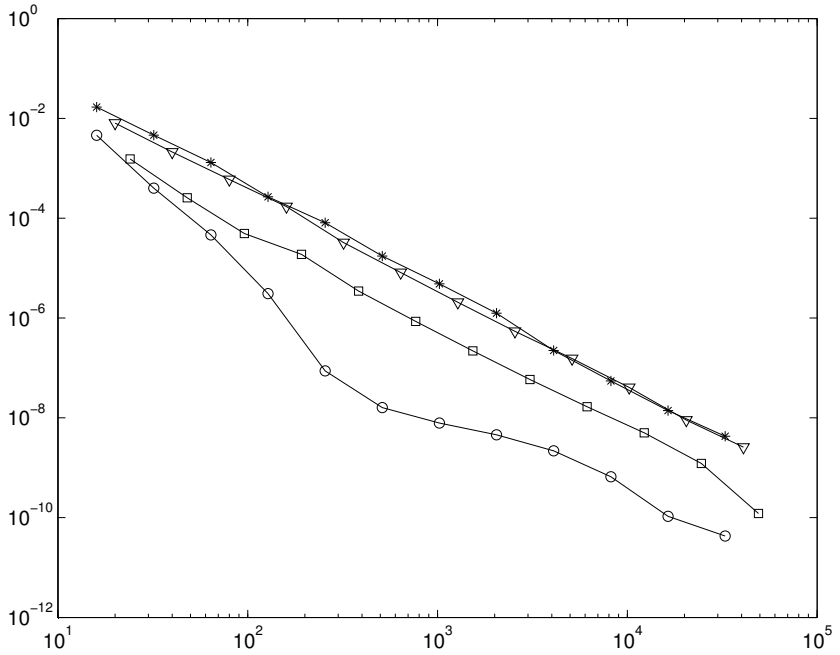

Figure 5.1: Problem 5.1. Absolute errors in u plotted against computational work (number of evaluations of g). StaggeredLF4 o-marks, SymmetricCO4 ∇-marks, RKN45 $*$-marks, RKN57 $\Box$-marks.

References

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

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

[3] M.P. Calvo, J.M. Sanz-Serna (1993), *The development of variable-step symplectic integrators, with application to the two-body problem*, SIAM J. Sci. Comput. 14, pp. 936 – 952.

[4] Lin-Yi Chou, P.W. Sharp (2000), *On order 5 symplectic Runge-Kutta-Nyström methods*, J. of Applied Mathematics & Decision Sciences 4, pp. 143–150.

[5] K. Dekker, J.G. Verwer (1984), *Stability of Runge-Kutta Methods for Stiff Nonlinear Differential Equations*. North-Holland, Amsterdam.

[6] Michelle Lynn Ghrist (2000), *High-Order Finite Difference Methods for Wave Equations*, PhD Thesis, University of Colorado, Boulder, USA.

[7] Michelle Ghrist, Bengt Fornberg, Tobin A. Driscoll (2000), *Staggered time integrations for wave equations*, SIAM. J. Numer. Anal. 38, pp. 718–741.

[8] E. Hairer, C. Lubich, G. Wanner (2002), *Geometric Numerical Integration*, Springer Series in Computational Mathematics, Vol. 31, Springer.

[9] R.I. McLachlan (1995), *On the numerical integration of ordinary differential equations by symmetric composition methods*, SIAM J. Sci. Comput. 16, pp. 151–168.

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

[11] J.M. Sanz-Serna, J.G. Verwer, W.H. Hundsdorfer (1987), *Convergence and order reduction of Runge-Kutta schemes applied to evolutionary problems in partial differential equations*, Numer. Math. 50, pp. 405–418.

[12] J.G. Verwer (2007), *On time staggering for wave equations*, CWI-report MAS-E0702 (preprint), to appear in Journal of Scientific Computing.