Convergence Rates for Adaptive Approximation of Ordinary Differential Equations

Kyoung-Sook Moon[1], **Anders Szepessy**[1,2], **Raúl Tempone**[1], **Georgios E. Zouraris**[3]

[1] Institutionen för Numerisk Analys och Datalogi, Kungl Tekniska Högskolan, S–100 44 Stockholm; e-mail: `moon@nada.kth.se`, `rtempone@nada.kth.se`

[2] Matematiska Institutionen, Kungl Tekniska Högskolan, S–100 44 Stockholm; e-mail: `szepessy@math.kth.se`

[3] Department of Mathematics, University of the Aegean, GR–832 00 Karlovassi, Samos, Greece, and IACM, FO.R.T.H, GR–711 10 Heraklion, Greece; e-mail: `zouraris@aegean.gr`

Summary   This paper constructs an adaptive algorithm for ordinary differential equations and analyzes its asymptotic behavior as the error tolerance parameter tends to zero. An adaptive algorithm, based on the error indicators and successive subdivision of time steps, is proven to stop with the optimal number, N, of steps up to a problem independent factor defined in the algorithm. A version of the algorithm with decreasing tolerance also stops with the total number of steps, including all refinement levels, bounded by $\mathcal{O}(N)$. The alternative version with constant tolerance stops with $\mathcal{O}(N \log N)$ total steps. The global error is bounded by the tolerance parameter asymptotically as the tolerance tends to zero. For a p-th order accurate method the optimal number of adaptive steps is proportional to the p-th root of the $L^{\frac{1}{p+1}}$ quasi-norm of the error density, while the number of uniform steps, with the same error, is proportional to the p-th root of the larger L^1-norm of the error density.

1 Introduction to Adaptive ODE Methods

This paper constructs an adaptive method, for approximation of ordinary differential equations, and analyzes its asymptotic behavior as the error tolerance parameter tends to zero. The algorithm is based on error indicators for the global discretization error of the form

$$\text{global error} = \sum_{\text{time steps}} \text{local error} \cdot \text{weight} + \text{higher order error}. \quad (1.1)$$

Consider a solution $X : [0, T] \to \mathbb{R}^d$ of a differential equation, with flux $a : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$,

$$\begin{aligned}
\frac{dX}{dt}(t) &= a(t, X(t)), \quad 0 < t \le T, \\
X(0) &= X_0,
\end{aligned} \quad (1.2)$$

and an approximation $\overline{X}$ of (1.2) by any numerical method, satisfying the same initial condition

$$\overline{X}(0) = X(0) = X_0 \quad (1.3)$$

with time steps

$$0 = t_0 < \cdots < t_N = T.$$

This work uses error estimates of the form (1.1) for the global error

$$g(X(T)) - g(\overline{X}(T)), \quad (1.4)$$

derived in [28], with a given general function $g : \mathbb{R}^d \to \mathbb{R}$, to construct adaptive time stepping methods. The function g is therefore included in the data of the problem, which the user specifies as in optimal control problems; i.e. the user provides the information to approximate the value of the objective function g. One example is to find the value of one component of the solution at the final time, e.g. $g(x) = x_1$.

The adaptive algorithm in this work can be viewed as a solution to the optimal control problem to minimize the number of time steps, with the error asymptotically bounded by the tolerance, for general numerical schemes of ordinary differential equations. The proposed adaptive algorithm yields an approximate solution to the optimal strategy

$$\left| \text{local error} \cdot \text{weight} \right| = \text{constant}. \quad (1.5)$$

The advantage with this approximation of functionals, g, of the solution and the optimal control approach to adaptive methods is that the weights can be computed with additional work which is of the

same order as that of solving for X. Alternative adaptive methods based on global error control of L^p-norms of the error require either more expensive computations of the weights, as compared to the work to compute X, for $d \gg 1$, or some a priori estimates of the weights, see [20, 13, 14, 23]. In particular the work [20, 13, 14, 23] focuses on the more important and much harder goal of guaranteed global error control, which is possible for certain differential equations allowing good a priori error estimates; this higher goal, not considered here, would clearly also justify more computational work. On the other hand, our algorithm treats roundoff errors, which often are neglected even in studies on guaranteed global error control.

Subsequent work extends the adaptive algorithms here to optimal control problems, see [32]. Our optimal control approach to adaptivity is inspired by the work [5–7] and [21] on finite element approximations. The work [15] studies discontinuous Galerkin approximation of optimal control problems with adaptive methods based on the alternative global error control in norms and arbitrary functionals.

There are numerous adaptive algorithms for ordinary and partial differential equations, e.g., [1], [3], [5–7], [12], [13], [18], [21], [27], [29], but the theoretical understanding of convergence rates of adaptive algorithms is not as well developed; there are however recent important contributions. DeVore studies in [10] the efficiency of adaptive approximation of functions, including wavelet expansions, based on smoothness conditions in Besov spaces. Inspired by this approximation result, Cohen, Dahmen and DeVore prove in [8] that a wavelet-based adaptive N-term approximation algorithm produces a solution with asymptotically optimal error $\mathcal{O}(N^{-s})$ in the energy norm for linear coercive elliptic problems. Our work connects DeVore's smoothness conditions to error densities for adaptive approximation of general ordinary differential equations. Adaptivity is also understood in the special case of integration, e.g., [17] shows that local error indicators give rigorous error bounds in an average probabilistic sense.

What is the right measure of convergence rates for adaptive algorithms? For a constant step size Δt, approximations with error $\mathcal{O}(\Delta t^p)$ require computational work with $\mathcal{O}(1/\Delta t)$ operations. The accuracy $\epsilon \equiv \mathcal{O}(\Delta t^p)$ is hence asymptotically determined by the number of steps $N = \mathcal{O}(1/\Delta t) = \mathcal{O}(\epsilon^{-1/p})$. This simple asymptotic complexity estimate, $\mathcal{O}(\epsilon^{-1/p})$, is one of the most basic and well used numerical analysis measures of the performance of approximations. Analogously, for adaptive methods, it seems natural to study the approximation error and the associated work, proportional to the number of steps, as the tolerance parameter tends to zero. For a p-th order

accurate method, the number of uniform steps to reach a given approximation error turns out to be proportional to the p-th root of the L^1-norm of the error density, defined by (local error $\cdot$ weight)$/\Delta t^{p+1}$, while the smallest number of adaptive steps is proportional to the p-th root of the smaller $L^{\frac{1}{p+1}}$ quasi-norm of the error density. These norms are therefore good measures of the convergence rates and define our optimal number of steps: Theorems 2.1, 2.4 and 2.5 in Section 2 prove that an adaptive algorithm stops with the optimal number of steps, N, up to a problem independent factor and the global error is asymptotically bounded by the tolerance times a problem independent factor, as the tolerance parameter tends to zero. The total number of time steps, including all refinement levels, can be bounded by the number of steps on the finest level times a problem independent factor, provided the tolerance in each refinement level decreases by a constant factor to guarantee that the number of steps increases at least by a given factor, see Theorem 2.7. Varying tolerance has the drawback that the final stopping tolerance is not a priori known; on the other hand, with constant tolerance, the total number of steps including all levels is bounded by the larger $\mathcal{O}(N \log N)$. The reports [33, 25] and [26] introduce adaptive algorithms for weak approximation of stochastic differential equations and partial differential equations, respectively, in the spirit of Section 2. The extensions of Theorems 2.1, 2.4 and 2.5 to stochastic and partial differential equations are straight-forward except for the convergence of the error density: to prove convergence of the error density for approximation of ordinary differential equations is simple, while the corresponding convergence result for stochastic and partial differential equations are subtle and require special techniques and new ideas, see [25] and [26].

The authors are not aware of any results on convergence rates and asymptotic work related to Theorem 2.1, 2.4, 2.5 and 2.7 for other algorithms to solve ordinary differential equations. One reason for this is that most adaptive algorithms are based on making a combination of the absolute and the relative local errors approximately constant, ignoring the weights, cf. [18], [34]. Although these algorithms in practice perform very well, a proof of the optimality of the mesh is lost; since the tolerance parameter measures only the local error, there is by (1.1) no explicit relation between this tolerance parameter and the global error. Many such algorithms also lack proofs of convergence of the approximations. One exception is the work [22, 31], which in particular proves the convergence of `ODE23` of `MATLAB` version 4.2 solving ordinary differential equations. Adaptivity based on the local errors, without the weights, has the clear advantage to

avoid the additional storage and work needed to compute the weight at many time levels. This additional storage is clearly a drawback. On the other hand many computer programs for the numerical solution of ordinary differential equations store the solution at all time levels for other reasons, e.g. for post processing. The use of dual functions is standard in optimal control theory and also well known for adaptive mesh control for ordinary and partial differential equations, see [2], [5], [7], [14], [20], [21], [36].

The literature on information based complexity, cf. [4], [30], [35], [37], discuss the efficiency of adaptive versus non-adaptive methods. A central result by Bakhvalov and Smolyak proves that, using a fixed number of functional evaluations, there is for each adaptive method a non-adaptive method which has as small maximal error as the adaptive method for approximation of linear functionals, $S : \mathcal{F} \to \mathbb{R}$, such as, e.g., $Sf = \int_0^1 f(t)dt$, with functions f in a convex symmetric subset $\mathcal{F}$ of a normed linear function space. A symmetric set is a set which contains $-f$, if f is in the set. A precise statement of the theorem is in Remark 2.10. Starting from Bakhvalov and Smolyak's result, the insightful review [30] includes discussion about when adaptive methods for integration and solution of ordinary and partial differential equations are useful. Our study differs from Bakhvalov and Smolyak's work in two important regards: Section 2 and 3 prove that an adaptive algorithm applied to a fixed differential equation (1.2) (and a fixed discretization method), uses a number of time steps which is asymptotically close to the optimal number to approximate with a given error tolerance, for the error (1.4), as the number of steps tends to infinity. In contrast, Bakhvalov and Smolyak analyze discretization methods based on the maximal error in a convex function set, with a fixed number of steps. The performance of the algorithm in Section 2 and 3 is not characterized by convex function sets; on the contrary, applied to integration, i.e., $a(t,x) = a(t)$ in (1.2), the estimate of the number of steps to approximate with error TOL in Section 2, using a p-th order accurate method, shows that adaptive integration is much more efficient than uniform steps, asymptotically as TOL $\to 0$, if

$$\|a^{(p)}\|_{L^{\frac{1}{p+1}}(0,T)} \quad \ll \quad \|a^{(p)}\|_{L^1(0,T)},$$

where $a^{(p)} \equiv d^p a/dt^p$ is the error density (non adaptive methods with non uniform steps would require some additional a priori information to improve over uniform steps). In particular, the functions which can be adaptively integrated, with given asymptotic behavior of the error

and number of steps, are characterized by the non convex set

$$\left\{ a \in C^p([0,T]) : \quad \|a^{(p)}\|_{L^{\frac{1}{p+1}}(0,T)} \le c \right\} \tag{1.6}$$

for a constant c. Our goal is to solve a problem to a certain accuracy with minimal asymptotic work by using appropriate adaptive time steps. We do not address the related problem to adaptively determine the order of the method and to determine implicit/explicit alternatives. The closely related problem of efficient adaptive and non adaptive approximation of functions, measured in L^q norms, has been characterized by DeVore [10] using Besov spaces, see Remark 2.11.

In conclusion, the main results are:

– a measure of convergence rates for adaptive approximation of ordinary differential equations;

– a general adaptive algorithm, where this work on ordinary differential equations is the basis for the extensions to stochastic and partial differential equations in [25, 26]; and

– a rigorous and simple analysis of convergence rates of an adaptive algorithm, where several related algorithms were successively improved to finally have both good numerical and theoretical results, with assumptions that are reasonable also in practice and not only for very small error tolerances.

The outline of the paper is: Section 2 describes and analyzes an adaptive algorithm; Section 3 presents numerical experiments based on the adaptive algorithm.

2 An Adaptive Algorithm

This section describes general properties of an adaptive algorithm. First we recall an expansion (1.1) of the global approximation error, derived in [28], and based on the local error and a variational principle. Then an adaptive algorithm is presented for problem (1.2). The algorithm chooses the number of time steps adaptively, by successively dividing time steps, to bound an approximation of the global error. The main result is that each refinement level in the algorithm decreases the maximal error indicator with at least a given factor until the algorithm stops with the optimal number of steps, up to a multiplicative constant factor which is independent of the problem (1.2). The true global error is then bounded by the tolerance times a similar problem independent factor, asymptotically as the tolerance tends to zero.

2.1 An Error Expansion

The adaptive algorithm we construct in this paper uses the error expansion (2.8) derived in [28], with computable leading order term based on approximate local errors and weights defined as follows.

Consider a p-th order accurate one step approximation $\overline{X}$, of X, written in the form

$$\overline{X}(t_n) = A(\overline{X}(t_{n-1}), \Delta t_n), \tag{2.1}$$

with time levels t_n and initial condition (1.3). The weights can then be approximated by

$$\overline{\Psi}_i(t_{n-1}) = \sum_{j=1}^{d} \partial_{x_i} A_j(\overline{X}(t_{n-1}), \Delta t_n)\overline{\Psi}_j(t_n),$$
$$\overline{\Psi}_i(T) = \partial_{x_i} g(\overline{X}(T)), \tag{2.2}$$

which yields a p-th order accurate approximation

$$\max_n |X(t_n) - \overline{X}(t_n)| + \max_n |\Psi(t_n) - \overline{\Psi}(t_n)| = \mathcal{O}((\max \Delta t)^p), \tag{2.3}$$

where $\Delta t_n = t_n - t_{n-1}$ and $\max \Delta t \equiv \max_n \Delta t_n$.

Let the local error e be defined by

$$e(t_n) \equiv \widetilde{X}(t_n) - \overline{X}(t_n), \tag{2.4}$$

where the local exact solution $\widetilde{X}$ satisfies, for each time step $(t_{n-1}, t_n]$,

$$\frac{d\widetilde{X}}{dt}(t) = a(t, \widetilde{X}(t)), \quad t_{n-1} < t \leq t_n,$$
$$\widetilde{X}(t_{n-1}) = \overline{X}(t_{n-1}). \tag{2.5}$$

We approximate the local error $e = \widetilde{X} - \overline{X}$ by replacing the unknown exact local solution $\widetilde{X}$ by an approximation $\overline{\overline{X}}$ of higher accuracy than $\overline{X}$, i.e., with smaller time steps or a higher order method in a higher precision. For smooth solutions X, the existence of the limits

$$\lim_{\Delta t \to 0} (\Delta t_n)^{-(p+1)} \left(\widetilde{X}(t_n) - \overline{X}(t_n) \right),$$
$$\lim_{\Delta t \to 0} (\Delta t_n)^{-(q+1)} \left(\widetilde{X}(t_n) - \overline{\overline{X}}(t_n) \right), \tag{2.6}$$

determines by Richardson extrapolation a constant γ, for $q \geq p$ cf. [9], such that

$$e(t_n) = \widetilde{X}(t_n) - \overline{X}(t_n) = \gamma \left(\overline{\overline{X}}(t_n) - \overline{X}(t_n) \right) + \mathcal{O}(\Delta t_n^{p+2}). \tag{2.7}$$

For instance there holds: $\gamma = 2^p/(2^p - 1)$ for $\overline{\overline{X}}$ computed with the half mesh size and $q = p$; and $\gamma = 1$ for $\overline{\overline{X}}$ computed with a higher order method $q > p$, see [18].

The global approximation error for the differential equation (1.2) then has the expansion

$$g(X(T)) - g(\overline{X}(T)) = \sum_{n=1}^{N} \left((\bar{e}(t_n), \overline{\Psi}(t_n)) + \mathcal{O}(\Delta t_n^{p+2}) \right) \quad (2.8)$$

$$= \sum_{n=1}^{N} \rho_n \Delta t_n^{p+1},$$

where

$$\rho_n \equiv \frac{(\bar{e}(t_n), \overline{\Psi}(t_n)) + \mathcal{O}(\Delta t_n^{p+2})}{\Delta t_n^{p+1}}$$

and $\bar{e}(t_n) \equiv \gamma \left(\overline{\overline{X}}(t_n) - \overline{X}(t_n) \right)$ is the approximation of the local error in (2.7).

2.2 Adaptive Step Size Control

Let us now motivate the optimal choice of steps

$$\left| \text{local error} \cdot \text{weight} \right| = \text{constant},$$

for approximation methods which have no essential constraint on the step sizes, such as one step methods (2.1). For the time steps $0 = t_0 < \cdots < t_N = T$, let the piecewise constant mesh function Δt be defined by

$$\Delta t(\tau) \equiv \Delta t_i \equiv t_i - t_{i-1} \quad \text{for} \quad \tau \in (t_{i-1}, t_i] \quad \text{and} \quad i = 1, \ldots, N.$$

Then the number of time steps that corresponds to a mesh Δt, for the interval $[0, T]$, can be defined by

$$N(\Delta t) \equiv \int_0^T \frac{1}{\Delta t(\tau)} d\tau. \quad (2.9)$$

Consider, for $\tau \in (t_{i-1}, t_i]$ and $i = 1, \ldots, N$, the piecewise constant function ρ, which measures the density of the global error from (2.8)

$$\rho(\tau) \equiv \rho_i \equiv \frac{(\bar{e}(t_i), \overline{\Psi}(t_i))}{\Delta t_i^{p+1}} + \mathcal{O}(\Delta t_i) \quad (2.10)$$

and its approximate counterpart $\bar{\rho}$, obtained from (2.8) with

$$\bar{\rho}(\tau) \equiv \bar{\rho}_i \equiv \text{sign}\left(\bar{e}(t_i), \overline{\Psi}(t_i)\right) \max\left(\frac{|\left(\bar{e}(t_i), \overline{\Psi}(t_i)\right)|}{\Delta t_i^{p+1}}, \delta\right) \quad (2.11)$$

where

$$\delta \equiv \text{TOL}^{\bar{\gamma}}, \quad 0 < \bar{\gamma} < \frac{1}{p+1}, \quad (2.12)$$

and $\text{sign}(x) = 1$ for $x \geq 0$ and -1 for $x < 0$. The constant $\delta > 0$ is motivated by the requirements that $\max \Delta t \to 0$ as $\text{TOL} \to 0$ and that the bounds for the error density in (2.22) hold, see Lemma 2.2.

It seems hard to use the sign of the error indicator for constructing the mesh, since with only two steps the error can be zero just by chance: let $\int_0^1 f(s)ds = 0$ be the integral of a continuous function where also $f(0) = f(1) = 0$. This integral can be computed by the Euler method without error for a very particular choice of just the two time steps $(0, \bar{s}), (\bar{s}, 1)$, with an interior point $\bar{s}$ satisfying $f(\bar{s}) = 0$, but any other choice of time steps gives in general very large errors. On the other hand, the cancellation of the error does not seem to be important in many cases, e.g. the Lorenz problem shows that the error only grows with a factor of two by using $|\bar{\rho}|$ instead of $\bar{\rho}$, see Remark 3.4. We choose to minimize the number of steps N in (2.9) under the more stringent constraint

$$\sum_{i=1}^{N} |\bar{\rho}_i| \Delta t_i^{p+1} = \int_0^T |\bar{\rho}(\tau)| \Delta t^p(\tau) d\tau = \text{TOL}. \quad (2.13)$$

This yields, with a standard application of a Lagrange multiplier, the optimal time steps Δt^* satisfying

$$|\bar{\rho}_i| \Delta t_i^{p+1} = constant \quad (2.14)$$

and

$$\Delta t^* \equiv \frac{\text{TOL}^{\frac{1}{p}}}{|\bar{\rho}|^{\frac{1}{p+1}}} \left(\int_0^T |\bar{\rho}(\tau)|^{\frac{1}{p+1}} d\tau\right)^{-\frac{1}{p}}. \quad (2.15)$$

This optimal choice gives $\text{TOL} = |\text{E}_\text{T}|$, where

$$\text{E}_\text{T} \equiv \sum_{i=1}^{N} \bar{\rho}_i \Delta t_i^{p+1}, \quad (2.16)$$

only for problems with positive density functions $\bar{\rho}$, since otherwise (2.16) and (2.13) may give $\text{TOL} \gg |\mathsf{E_T}|$. To use the sign of the density in an optimal way is not considered in this work.

The goal of the adaptive algorithm described below is to construct a partition Δt of $[0, T]$ such that

$$|\bar{\rho}_i| \Delta t_i^{p+1} \approx \frac{\text{TOL}}{N}, \quad \forall\, i = 1, \ldots, N, \qquad (2.17)$$

which is an approximation of the optimal (2.14). To achieve (2.17) let $s_1 \approx 1$ be a given constant, start with an initial partition $\Delta t[1]$ and then specify iteratively a new partition $\Delta t[k + 1]$, from $\Delta t[k]$, using the following division strategy: for $i = 1, 2, \ldots, N[k]$, let

$$\bar{r}_i[k] \equiv |\bar{\rho}_i[k]|(\Delta t_i[k])^{p+1}, \qquad (2.18)$$

and

> **if** $\quad \bar{r}_i[k] > s_1 \dfrac{\text{TOL}}{N[k]} \quad$ **then**
>
> $\quad$ divide $\Delta t_i[k]$ into M uniform substeps
>
> **else**
>
> $\quad$ let the new step be the same as the old
>
> **endif**
$$\tag{2.19}$$

where M is a given integer greater than 1. With this division strategy, it is natural to use the stopping criterion:

$$\textbf{if}\quad \left(\max_{1 \leq i \leq N[k]} \bar{r}_i[k] \leq S_1 \frac{\text{TOL}}{N[k]} \right) \quad \textbf{then}\quad \text{stop.} \qquad (2.20)$$

Here S_1 is a given constant, with $S_1 > s_1 \approx 1$, determined more precisely as follows: we want the maximal error indicator to decay quickly to the stopping level $S_1 \text{TOL}/N$, but when almost all $\bar{r}_i$ satisfy $\bar{r}_i \leq s_1 \frac{\text{TOL}}{N}$, the reduction of the error may be slow. Theorem 2.1 shows that slow reduction is avoided if S_1 satisfies (2.23). Refinements by subdivision related to (2.19) is standard in adaptive algorithms for partial differential equations, cf. [6], but the stopping (2.20) is not. We have tested several alternative stopping rules, such as the well known $|\mathsf{E_T}| \leq \text{TOL}$. It turns out that the stopping condition (2.20) yields more accurate error estimates both theoretically and computationally.

The remainder of this section analyzes in three theorems the adaptive algorithm based on (2.17) with respect to stopping, accuracy

and efficiency. In order to analyze the decay of the maximal error indicator, it is useful to understand the variation of the density $\bar{\rho}$ at different refinement levels. In particular we will consider a time step $(t_{i-1}, t_i][k]$ and its parent on a previous refinement level, $\text{parent}(i, k)$, with the corresponding error density $\bar{\rho}(t_i)[\text{parent}(i, k)]$. Since by (2.29), $\text{TOL} \to 0+$ implies that $\max \Delta t \to 0$, there is a limit, $\widetilde{\rho}$, of ρ using $\overline{\Psi} \to \Psi$ by (2.3) and $\bar{e}/\Delta t^{p+1} - e/\Delta t^{p+1} \to 0$ by (2.6, 2.7) as $\max \Delta t \to 0$, thus

$$|\bar{\rho}| \to |\widetilde{\rho}|, \quad \text{as } \max \Delta t \to 0. \tag{2.21}$$

A consequence of (2.21) as $\text{TOL} \to 0+$, and (2.10, 2.11) is that for all time steps $(t_{i-1}, t_i][k]$ and all refinement levels k there exists a constant $c = c(t_i)$, close to 1 for sufficiently refined meshes, such that the error density satisfies

$$c \leq \left| \frac{\bar{\rho}(t_i)[\text{parent}(i, k)]}{\bar{\rho}(t_i)[k]} \right| \leq c^{-1},$$

$$c \leq \left| \frac{\bar{\rho}(t_i)[k-1]}{\bar{\rho}(t_i)[k]} \right| \leq c^{-1}, \tag{2.22}$$

provided $\max_{n,k} \Delta t_n[k]$ is sufficiently small. Section 3 verifies the condition (2.22) computationally for some examples; in particular the Lorenz problem in Figure 3.3 shows that c^{-1} for the maximal error indicator is close to 1, while $\max_i c(t_i)^{-1}$ can be very large. Note that the condition (2.22) also implies a related constraint on the optimal mesh, see Remark 2.3.

Theorem 2.1 (Stopping) *Suppose the adaptive algorithm uses the strategy (2.18)-(2.20). Assume that c satisfies (2.22) for the time steps corresponding to the maximal error indicator on each refinement level, and that*

$$S_1 \geq \frac{M}{c} s_1, \quad 1 > \frac{c^{-1}}{M^{p+1}}. \tag{2.23}$$

Then each refinement level either decreases the maximal error indicator with the factor

$$\max_{1 \leq i \leq N[k+1]} \bar{r}_i[k+1] \leq \frac{c^{-1}}{M^{p+1}} \max_{1 \leq i \leq N[k]} \bar{r}_i[k], \tag{2.24}$$

or stops the algorithm.

Proof. There is a $t^* \in [0, T]$ giving the maximal error indicator value

$$\bar{r}(t^*)[k+1] = \max_{1 \leq i \leq N[k+1]} \bar{r}_i[k+1]$$

on refinement level $k+1$. The corresponding indicator $\bar{r}(t^*)[k]$, on the previous level, satisfies precisely one of the following three statements

$$\bar{r}(t^*)[k] \leq \frac{s_1 \text{TOL}}{N[k]}, \tag{2.25}$$

$$\frac{s_1 \text{TOL}}{N[k]} < \bar{r}(t^*)[k] \leq M^{p+1} \frac{s_1 \text{TOL}}{N[k]}, \tag{2.26}$$

$$\bar{r}(t^*)[k] > M^{p+1} \frac{s_1 \text{TOL}}{N[k]}. \tag{2.27}$$

If (2.25) holds the time step containing t^* is not divided on level $k+1$ and by (2.22)

$$\bar{r}(t^*)[k+1] \leq \frac{c^{-1} s_1 \text{TOL}}{N[k]}. \tag{2.28}$$

Condition (2.23) and the bound $N[k+1] \leq MN[k]$ imply $\frac{S_1 \text{TOL}}{N[k+1]} \geq \frac{c^{-1} s_1 \text{TOL}}{N[k]}$, which together with (2.28) show that the algorithm stops at level $k+1$ if (2.25) holds.

Similarly, if (2.26) holds, the time step containing t^* is divided on level $k+1$, so that $\bar{r}(t^*)[k+1] \leq \frac{c^{-1} s_1 \text{TOL}}{N[k]}$ again and consequently the algorithm stops at level $k+1$.

Finally if (2.27) holds, the time step containing t^* is divided and by (2.22)

$$\bar{r}(t^*)[k+1] \leq \frac{c^{-1}}{M^{p+1}} \bar{r}(t^*)[k] \leq \frac{c^{-1}}{M^{p+1}} \max_{1 \leq i \leq N[k]} \bar{r}_i[k],$$

which proves the theorem.   □

Let us verify that the choice (2.12) of δ implies that $\max \Delta t \to 0$ and that c is close to 1 in (2.22) for sufficiently refined meshes.

Lemma 2.2 *Suppose* (2.8), (2.10-2.12), *and* (2.21) *hold, then*

$$\lim_{\text{TOL} \to 0+} \max_t \Delta t(t)[J] = 0, \tag{2.29}$$

for the final mesh J, and

$$\left| \frac{|\bar{\rho}(t_i)[\text{parent}(i,k)]|}{|\bar{\rho}(t_i)[k]|} - 1 \right| = \mathcal{O}\left(\text{TOL}^{\frac{1-(p+1)\bar{\gamma}}{p}} \right),$$

$$\left| \frac{|\bar{\rho}(t_i)[k-1]|}{|\bar{\rho}(t_i)[k]|} - 1 \right| = \mathcal{O}\left(\text{TOL}^{\frac{1-(p+1)\bar{\gamma}}{p}} \right).$$

Proof. When the algorithm stops the error indicators satisfy the bound

$$|\bar{\rho}_i| \Delta t_i^{p+1} \leq \frac{S_1 \mathrm{TOL}}{N}, \quad \text{for all } i. \tag{2.30}$$

Consequently we have by (2.12)

$$\delta \max \Delta t^{p+1} \leq \frac{S_1 \mathrm{TOL}}{N},$$

which proves (2.29):

$$\max \Delta t^p \leq \frac{S_1 \mathrm{TOL}}{\delta(N \max \Delta t)} \leq \frac{S_1 \mathrm{TOL}^{1-\bar{\gamma}}}{T}.$$

The definition (2.11) implies $|\bar{\rho}| = \max(|\widetilde{\rho}| + \mathcal{O}(\max \Delta t), \delta)$, where $\widetilde{\rho}$ is the limit of $\bar{\rho}$ obtained in (2.21). Therefore, we have

$$\left| \frac{|\bar{\rho}(t_i)[k-1]|}{|\bar{\rho}(t_i)[k]|} - 1 \right| \leq \frac{\mathcal{O}(\max \Delta t[k])}{\delta} = \mathcal{O}(\mathrm{TOL}^{\frac{1-\bar{\gamma}}{p}-\bar{\gamma}}).$$

The same estimate for $\frac{|\bar{\rho}(t_i)[\mathrm{parent}(i,k)]|}{|\bar{\rho}(t_i)[k]|}$ finishes the proof. □

Remark 2.3 The error density condition (2.22) also implies constraints on the optimal mesh; for instance, $M = 2$ and the assumption $\frac{1}{2}(\bar{\rho}_i[k] + \bar{\rho}_{i+1}[k]) = \bar{\rho}(t_i)[k-1]$ shows that

$$2c - 1 \leq \left| \frac{\bar{\rho}_{i+1}[k]}{\bar{\rho}_i[k]} \right| \leq 2c^{-1} - 1. \tag{2.31}$$

□

2.3 Accuracy of the Adaptive Algorithm

The adaptive algorithm guarantees that the estimate of the global error is bounded by a given error tolerance, TOL. An important question is whether or not the true global error is bounded by TOL asymptotically. Using the upper bound (2.20) of the error indicators and the convergence of ρ and $\bar{\rho}$ in (2.8, 2.10, 2.11, 2.21), the global error has the estimate

Theorem 2.4 (Accuracy) *Suppose that the assumptions of Lemma 2.2 hold. Then the adaptive algorithm* (2.18)-(2.20) *satisfies*

$$\limsup_{\mathrm{TOL} \to 0+} \left(\mathrm{TOL}^{-1} |g(X(T)) - g(\overline{X}(T))| \right) \leq S_1. \tag{2.32}$$

Proof. When the adaptive algorithm stops, (2.8), (2.10) and (2.20) imply

$$
\begin{aligned}
\text{TOL}^{-1}|g(X(T)) - g(\overline{X}(T))| &\le \text{TOL}^{-1} \sum_{i=1}^{N} \Delta t_i^p \int_{t_{i-1}}^{t_i} |\rho(\tau)| d\tau \\
&\le \text{TOL}^{-1} \left(S_1 \frac{\text{TOL}}{N} \right)^{\frac{p}{p+1}} \sum_{i=1}^{N} \int_{t_{i-1}}^{t_i} \frac{|\rho(\tau)|}{|\bar\rho(\tau)|^{\frac{p}{p+1}}} d\tau.
\end{aligned}
\tag{2.33}
$$

Rewrite the inequality (2.20) as

$$
|\bar\rho|^{\frac{1}{p+1}} \le \left(S_1 \frac{\text{TOL}}{N} \right)^{\frac{1}{p+1}} \frac{1}{\Delta t_i},
$$

integrate both sides and use the definition (2.9) to obtain

$$
N^{-\frac{p}{p+1}} \le (S_1 \text{TOL})^{\frac{1}{p+1}} \frac{1}{\int_0^T |\bar\rho(\tau)|^{\frac{1}{p+1}} d\tau}.
$$

Apply this to the right hand side of (2.33) to get

$$
\text{TOL}^{-1}|g(X(T)) - g(\overline{X}(T))| \le S_1 \frac{\int_0^T |\rho(\tau)|/|\bar\rho(\tau)|^{\frac{p}{p+1}} d\tau}{\int_0^T |\bar\rho(\tau)|^{\frac{1}{p+1}} d\tau}.
\tag{2.34}
$$

Since by Lemma 2.2 we have $\max \Delta t \to 0$ and consequently ρ and $\bar\rho$ converge to $\widetilde\rho$ as $\text{TOL} \to 0+$, the fraction in (2.34) converges to 1 by the Lebesgue dominated convergence theorem, which proves (2.32).
$\square$

2.4 Efficiency of the Adaptive Algorithm

An important issue for the adaptive method is its efficiency: we want to determine a partition with as few time steps as possible providing the desired accuracy. The definition (2.9) and the optimality condition (2.15) shows that the number of optimal adaptive steps, N^{opt}, satisfies

$$
N^{\text{opt}} = \int_0^T \frac{1}{\Delta t^*(\tau)} d\tau = \frac{1}{\text{TOL}^{\frac{1}{p}}} \left(\int_0^T |\bar\rho[k](\tau)|^{\frac{1}{p+1}} d\tau \right)^{\frac{p+1}{p}},
$$

i.e.,

$$
N^{\text{opt}} = \frac{1}{\text{TOL}^{\frac{1}{p}}} \|\bar\rho\|_{L^{\frac{1}{p+1}}}^{\frac{1}{p}}.
\tag{2.35}
$$

Here $p > 0$ is the order of accuracy of the approximate solution and $\| \cdot \|_{L^{\frac{1}{p+1}}}$ is the quasi-norm defined by

$$\|f\|_{L^{\frac{1}{p+1}}} \equiv \left(\int_0^T |f(x)|^{\frac{1}{p+1}} dx \right)^{p+1}.$$

On the other hand, for the uniform steps $\Delta t = constant$, the number of steps, N^{uni}, to achieve $\sum_{i=1}^N |\bar{\rho}_i| \Delta t_i^{p+1} = \mathrm{TOL}$ becomes

$$N^{\mathrm{uni}} = \int_0^T \frac{1}{\Delta t(\tau)} d\tau = \frac{T}{\mathrm{TOL}^{\frac{1}{p}}} \left(\int_0^T |\bar{\rho}[k](\tau)| d\tau \right)^{\frac{1}{p}},$$

i.e.,

$$N^{\mathrm{uni}} = \frac{T}{\mathrm{TOL}^{\frac{1}{p}}} \|\bar{\rho}\|_{L^1}^{\frac{1}{p}}. \qquad (2.36)$$

Hence, the number of uniform steps is measured in the L^1-norm and the optimal number of steps is measured in the $L^{\frac{1}{p+1}}$ quasi-norm. Jensen's inequality implies $\|f\|_{L^{\frac{1}{p+1}}} \leq T^p \|f\|_{L^1}$, therefore an adaptive method may use fewer time steps than the uniform step size method, see Remarks 2.10 and 2.11, (1.6) and Example 3.2.

The following theorem uses a lower bound of the error indicators, obtained from the stopping condition (2.20) and the ratio of the error density (2.22), to show that the algorithm (2.18)-(2.20) generates a mesh which is optimal, up to a multiplicative constant.

Theorem 2.5 (Efficiency) *Assume that $c = c(t)$ satisfies (2.22) for all time steps at the final refinement level, that all initial time steps have been divided when the algorithm stops, and that the assumptions of Lemma 2.2 hold. Then there exists a constant $C > 0$, bounded by $(\frac{M^{p+1}}{s_1})^{\frac{1}{p}}$, such that the final number of adaptive steps N, of the algorithm (2.18)-(2.20), satisfies*

$$\mathrm{TOL}^{\frac{1}{p}} N \leq C \, \|\frac{\bar{\rho}}{c}\|_{L^{\frac{1}{p+1}}}^{\frac{1}{p}} \leq C \left(\max_{0 \leq t \leq T} c(t)^{-\frac{1}{p}} \right) \|\bar{\rho}\|_{L^{\frac{1}{p+1}}}^{\frac{1}{p}}, \qquad (2.37)$$

and $\|\bar{\rho}\|_{L^{\frac{1}{p+1}}} \to \|\widetilde{\rho}\|_{L^{\frac{1}{p+1}}}$, asymptotically as $\mathrm{TOL} \to 0+$.

The Lorenz problem in Section 3 gives an example where the average $\|\frac{\bar{\rho}}{c}\|_{L^{\frac{1}{p+1}}} / \|\bar{\rho}\|_{L^{\frac{1}{p+1}}} \approx 1$ while $\max c^{-1} \approx 10^5$. Therefore the first bound in (2.37) yields a good estimate $N/N^{\mathrm{opt}} \approx 1$, but the second inequality yields the less accurate estimate $N/N^{\mathrm{opt}} \lesssim 10$, see Section 3. In fact, the main guideline for our construction of adaptive algorithms has been to find an algorithm for which convergence rates can be derived based on assumptions that are also satisfied in practice.

Proof. When the adaptive algorithm stops, on level k, the error indicators satisfy the upper bound

$$\bar{r}_i[k] = (|\bar{\rho}(t_i)|\Delta t_i^{p+1})[k] \leq \frac{S_1 \text{TOL}}{N[k]}.$$

By assumption, each time step $(t_{i-1}, t_i)[k]$ has a parent on a previous level, $\text{parent}(i, k)$ (not necessary the previous level $k-1$), which was divided. Therefore the indicators of the parent time steps satisfy the lower bound

$$
\begin{aligned}
&|\bar{\rho}(t_i)[\text{parent}(i,k)]|M^{p+1}\Delta t(t_i)^{p+1}[k] \\
&= (|\bar{\rho}(t_i)|\Delta t(t_i)^{p+1})[\text{parent}(i,k)] \\
&> \frac{s_1 \text{TOL}}{N[\text{parent}(i,k)]} \\
&\geq \frac{s_1 \text{TOL}}{N[k]}.
\end{aligned}
$$

The estimate on the number of steps now follows by relating the error indicators to the lower bounds of their parents:

$$
\begin{aligned}
\Delta t(t_i)^{p+1}[k] &> \frac{s_1 \text{TOL}}{N[k]} \frac{1}{M^{p+1}} \frac{1}{|\bar{\rho}(t_i)[\text{parent}(i,k)]|} \\
&\geq \frac{s_1 \text{TOL}}{N[k]M^{p+1}} \frac{c}{|\bar{\rho}(t_i)[k]|}.
\end{aligned}
$$

This and (2.9) imply

$$N[k] = \int_0^T \frac{dt}{\Delta t(t)[k]} < \frac{(N[k])^{\frac{1}{p+1}} M}{(s_1 \text{TOL})^{\frac{1}{p+1}}} \int_0^T \left|\frac{\bar{\rho}}{c}\right|^{\frac{1}{p+1}} dt$$

which together with Hölder's inequality proves the theorem

$$
\begin{aligned}
N[k] &\leq \left(\frac{M^{p+1}}{s_1}\right)^{\frac{1}{p}} \left(\frac{1}{\text{TOL}}\right)^{\frac{1}{p}} \left\|\frac{\bar{\rho}}{c}\right\|_{L^{\frac{1}{p+1}}}^{\frac{1}{p}} \\
&\leq \left(\frac{M^{p+1}}{s_1}\right)^{\frac{1}{p}} \left(\frac{1}{\text{TOL}}\right)^{\frac{1}{p}} (\|c^{-1}\|_{L^\infty} \|\bar{\rho}\|_{L^{\frac{1}{p+1}}})^{\frac{1}{p}}.
\end{aligned}
$$

$\square$

2.5 Implementation of the Adaptive Algorithm

This subsection presents a detailed implementation, called MSTZ, of the adaptive algorithm (2.18)-(2.20). The division strategy (2.19) is applied iteratively until the approximate solution is sufficiently resolved, in other words, until the approximate error density $\bar{\rho}$ and the time steps satisfy the stopping criteria (2.20):

Initialization. The user chooses
   1. an initial error tolerance, TOL,
   2. a number, $N[1]$, of initial uniform steps $\Delta t[1]$ for $[0, T]$,
   3. an integer number, M, of uniform subdivisions of each refined time step, and
   4. a number, s_1, in (2.19) and a rough estimate of c in (2.22) to compute S_1 using (2.23).
   Set the iteration number k to 0.

Step I. Increment the iteration number k by 1. For $n = 1, \ldots, N[k]$, compute the approximation $\overline{X}(t_n)$ of (1.2) using a p-th order accurate numerical method (2.1), and to obtain the local error, compute the approximate local exact solution $\overline{\overline{X}}(t_n)$ of (2.5) using a higher accuracy than for $\overline{X}(t_n)$. Compute the approximation of the local error $\bar{e}(t_n)$ by (2.7) and the approximate weight $\overline{\Psi}(t_n)$, for $n = N[k], \ldots, 1$, using the p-th order accurate method (2.2).

Step II. If (a local roundoff error condition, (2.25) or (2.26) in [28], holds) **then** terminate the program due to too large roundoff error

elseif $\left(\max_{1 \leq i \leq N[k]} \bar{r}_i[k] \leq \frac{S_1 \mathrm{TOL}}{N[k]} \right)$ **then** stop the program

else
   do for all time steps $i = 1, \ldots, N[k]$

$$\mathbf{if} \quad \left(\bar{r}_i[k] > s_1 \frac{\mathrm{TOL}}{N[k]} \right) \quad \mathbf{then}$$

$$\Delta t(t)[k+1] = \frac{\Delta t_i[k]}{M}, \quad t_{i-1}[k] < t \leq t_i[k],$$

   else
$$\Delta t(t)[k+1] = \Delta t_i[k], \quad t_{i-1}[k] < t \leq t_i[k],$$
   endif

   enddo
   go to Step I.
endif

2.6 Decreasing Tolerance

This subsection studies an adaptive algorithm allowing the tolerance to decrease slightly as the mesh is refined. The decreasing tolerance is motivated by efficiency –the efficiency of the algorithm depends on the total work including all refinement levels. If the number of elements in each refinement iteration increases only very slowly, the total work becomes proportional to the product of the number of steps in the finest mesh times the number of refinement levels. The condition (2.15) shows that the number of refined levels, J, satisfies

$$\min \Delta t = M^{-J} T / N[1] = \mathcal{O}(\text{TOL}^{1/p}). \qquad (2.38)$$

A relation $\min \Delta t = \mathcal{O}(\text{TOL}^{\alpha})$, $\alpha > 0$, still holds for many singular densities, as in Example 3.2. Therefore, $J = \mathcal{O}(\frac{1}{p} \log(\text{TOL}^{-1})) \simeq \log N$, so that the total number of steps for the algorithm (2.18-2.20) would be essentially bounded by

$$N \log N. \qquad (2.39)$$

A more efficient refinement algorithm is obtained by successively decreasing the tolerance, $\text{TOL}[k + 1] < \text{TOL}[k]$, in each refinement so that

$$\frac{N[k]}{N[k + 1]} \leq \bar{c} < 1 \qquad (2.40)$$

always holds. The condition (2.40) would imply that the total number of steps satisfy

$$\sum_{k=1}^{J} N[k] \leq \frac{N[J]}{1 - \bar{c}}. \qquad (2.41)$$

Therefore, a slightly decreasing tolerance may be more efficient than a constant tolerance, which yields the total work (2.39). Including the assumption

$$c' \leq \frac{\text{TOL}[k + 1]}{\text{TOL}[k]} \leq 1 \qquad (2.42)$$

and replacing c by $c'c$ in (2.23) directly generalizes Theorems 2.1, 2.4 and 2.5 to slightly varying tolerance, where TOL in (2.32) and (2.37) then denotes the final stopping tolerance. However, an unattractive consequence of varying tolerance is that the stopping tolerance becomes a priori uncertain, see Remark 2.6 and Theorem 2.7.

Remark 2.6 A decreasing tolerance is useful if there are few steps with their error indicators, $\bar{r}_i$, in the set $(s_1 \text{TOL}/N, \infty)$. To include a decreasing tolerance, modify the algorithm by adding the command "Set $\mathtt{V} = 0$" in the end of **Step I** and replace the statement " **go to Step I**" after **enddo** in the end of **Step II** by:

if $(N[k]/N[k+1] > \bar{c} \,\&\mathtt{V} = 0)$, **then**

    $\text{TOL} \equiv \text{TOL}[k](1 - \frac{\bar{c}^{-1}-1}{M-1})$, $\mathtt{V} = 1$ and **go to Step II**,

else

    go to Step I.

endif

Include in the initialization also a choice of the factor $\bar{c}$ to increase the number of steps in (2.40).

    Assume that the set $(c' s_1 \text{TOL}/N, s_1 \text{TOL}/N]$ contains a fraction $c''N$ of the steps, where $M^{-p} < c' < 1$; for instance, if the error indicators, $\bar{r}_i$, are uniformly distributed in $[0, s_1 \text{TOL}/N]$, with a negligible part outside of this set, there holds $c'' = 1 - c'$, which yields $\bar{c} = \frac{1}{1+c''(M-1)} = \frac{1}{1+(1-c')(M-1)}$ and motivates $c' = 1 - \frac{\bar{c}^{-1}-1}{M-1}$ in the algorithm. A refinement approximately maps the set

$$(c' s_1 \text{TOL}/N, s_1 \text{TOL}/N]$$

to

$$(c' s_1 \text{TOL}/(NM^{p+1}), s_1 \text{TOL}/(NM^{p+1})].$$

Then the next refinement continues with essentially a similar distribution of the error indicators, provided c' is not too small. When the algorithm stops, the final tolerance satisfies

$$\text{TOL}[0] \geq \text{TOL}[J] \geq \text{TOL}[0](c')^J = \text{TOL}^{1+\mathcal{O}(\frac{|\log c'|}{p})},$$

which for c' close to 1 is only a slight change. $\quad\square$

    Let us now show that the total number of steps can be bounded by a constant times the number of steps in the finest mesh, in the case of decreasing tolerance. The proof uses that the tolerance decreases sufficiently, which simplifies the analysis. A more refined study, with less demanding assumptions on the tolerance, following the idea in Remark 2.6 would need deeper understanding of the distribution of the error indicators $\bar{r}_i$. In contrast to the basic Theorems 2.1, 2.4 and 2.5, the following result has the drawback that it uses a uniform bound in (2.22) which yields a condition, on c', that in practice can be too restrictive although it seems reasonable for very small tolerances. The proof is also more complicated and less natural than the previous proofs.

Theorem 2.7 *The total number of steps satisfies the bound*

$$\sum_{k=1}^{J} N[k] = \mathcal{O}(N[J]),$$

for a variant of MSTZ *where all levels have decreasing tolerance*

$$\text{TOL}[k+1] = \text{TOL}[k]c'$$

satisfying $0 < c' < c$, *provided all initial steps are divided,* $S_1 \geq s_1 M/(cc')$ *and (2.22) holds uniformly for all time steps.*

Proof. Let $s_2 \equiv s_1 c/(c' M^{p+1})$ and $\mathcal{N}_0[k] \equiv \{i : s_2 \text{TOL}[k]/N[k] \leq \bar{r}_i[k] \leq s_1 \text{TOL}[k]/N[k]\}$. We shall first show that $\min_{n,k}(\bar{r}_n N/\text{TOL})[k] \geq s_2$. Assume first that $\min_n \bar{r}_n[k] > s_1 \text{TOL}/N[k]$, then all time steps are divided on level $k+1$ and by (2.22)

$$\begin{aligned}
\bar{r}(t_n)[k+1] &= (|\rho_n|\Delta t_n^{p+1})[k+1] \\
&\geq c|\rho(t_n)[k]|\frac{\Delta t(t_n)^{p+1}[k]}{M^{p+1}} \\
&= \frac{c}{M^{p+1}}\bar{r}(t_n)[k] \\
&> \frac{cs_1 \text{TOL}[k]}{M^{p+1}N[k]}
\end{aligned}$$

therefore

$$\begin{aligned}
\min_n \bar{r}_n[k+1] &> \frac{cs_1 \text{TOL}[k+1]}{c' M^{p+1}N[k+1]} \\
&= \frac{s_2 \text{TOL}[k+1]}{N[k+1]}.
\end{aligned}$$

Then if $n \in \mathcal{N}_0[k]$ the time step Δt_n is not divided on level $k+1$ so that

$$\begin{aligned}
\bar{r}_n[k+1] &\geq c\,\bar{r}_n[k] \\
&\geq c\,s_2\,\text{TOL}[k]/N[k] \\
&\geq \frac{c}{c'}\,s_2\,\text{TOL}[k+1]/N[k+1] \\
&> s_2\,\text{TOL}[k+1]/N[k+1].
\end{aligned}$$

Therefore we conclude, by induction, that the error indicators satisfy

$$\min_{n,k}\left(\frac{\bar{r}_n N}{\text{TOL}}\right)[k] \geq s_2.$$

The next step is show that at most m consecutive levels can have the slow increase $N[k]/N[k+1] > \bar{c}$. This will imply that the total number of steps is bounded by a constant times the final number of steps. Assume the contrary that

$$\frac{N[k]}{N[k+1]} > \bar{c}, \quad k = K, \ldots, K + m, \tag{2.43}$$

where m and $\bar{c}$ satisfy

$$\frac{(c')^m}{c} < \frac{s_2}{s_1}, \tag{2.44}$$

$$1 < \bar{c}^{-1} < M^{1/(m+1)}, \tag{2.45}$$

and let $N_0[k] \equiv \#\mathcal{N}_0[k]$ and $N_+ \equiv N - N_0$. The condition (2.43) and

$$N[k+1] = N_0[k] + MN_+[k]$$

show that the number of divided steps, $N_+[k]$, satisfies

$$N_+[k] < \frac{\bar{c}^{-1} - 1}{M - 1} N[k]. \tag{2.46}$$

The tolerance decreases, so that after m levels the dividing barrier is

$$s_1 \mathrm{TOL}[K + m]/N[K + m] < (c')^m s_1 \mathrm{TOL}[K]/N[K].$$

All elements in $\mathcal{N}_0[K]$ must have been divided after m levels, since if they have not all been divided some error indicators are larger than $cs_2 \mathrm{TOL}[K]/N[K]$ and condition (2.44) gives the contradiction

$$s_1 \frac{\mathrm{TOL}[K + m]}{N[K + m]} < (c')^m s_1 \frac{\mathrm{TOL}[K]}{N[K]} < cs_2 \frac{\mathrm{TOL}[K]}{N[K]}.$$

Dividing of all steps in $\mathcal{N}_0[K]$ shows that $N_0[K]$ must be smaller than the sum of divided steps

$$N_0[K] \leq \sum_{j=1}^{m} N_+[K + j] \tag{2.47}$$

which also leads to a contradiction, since by (2.46)

$$N_0[K] = N[K] - N_+[K] > \frac{M - \bar{c}^{-1}}{M - 1} N[K]$$

and by combining (2.46) and (2.43)

$$
\begin{aligned}
N_+[K+j] &< \frac{\bar{c}^{-1}-1}{M-1}N[K+j] \\
&< \frac{\bar{c}^{-1}-1}{M-1}\bar{c}^{-1}N[K+j-1] \\
&< \frac{\bar{c}^{-1}-1}{M-1}\bar{c}^{-j}N[K],
\end{aligned}
$$

so that by the assumption (2.45)

$$
N_0 - \sum_{j=1}^{m} N_+[K+j] > \frac{M-\bar{c}^{-m-1}}{M-1}N[K] > 0,
$$

which contradicts (2.47). Hence, the number of consecutive levels, where $N[k]/N[k+1] > \bar{c}$, must be smaller than $m+1$ and therefore

$$
\sum_{k=1}^{J} N[k] \leq \frac{mN[J]}{1-\bar{c}} = \mathcal{O}(N[J]).
$$

$\square$

2.7 Remarks

We end this section with four remarks on adaptive dividing and merging of steps, individual Δt for each component of the solution, the statement of Bakhvalov and Smolyak's theorem discussed on the introduction, and Besov spaces for adaptive integration.

Remark 2.8 The work [25] generalizes the present algorithm based only on dividing to also include merging of steps. These two adaptive algorithms perform similarly and analogous theoretical results are proved. A theoretical advantage without merging is that stopping requires (2.22) only at the maximal error indicator on each level and that fewer parameters are used. The dividing-merging adaptive algorithm takes the form: for $i = 1, 2, \ldots, N[k]$ let

$$
\bar{r}_i[k] \equiv |\bar{\rho}_i[k]|(\Delta t_i[k])^{p+1}
$$

and

$$\textbf{if} \quad \left(\bar{r}_i[k] > s_1 \frac{\text{TOL}}{N} \right) \quad \textbf{then}$$

divide $\Delta t_i[k]$ into M uniform substeps,

$$\textbf{elseif} \quad \max \left(\bar{r}_i[k], \bar{r}_{i+1}[k] \right) < s_2 \frac{\text{TOL}}{N}, \quad \textbf{then}$$

merge $\Delta t_i[k]$ and $\Delta t_{i+1}[k]$ into one step
and increase i by 1,

$\textbf{else}$ let the new step be the same as the old.

$\textbf{endif}$

With this the dividing and merging strategy it is natural to use the following stopping criteria:

$$\textbf{if} \quad \left(\bar{r}_i[k] \leq S_1 \frac{\text{TOL}}{N}, \quad \forall \, i = 1, \ldots, N \right) \textbf{and}$$

$$\left(\max \left(\bar{r}_i[k], \bar{r}_{i+1}[k] \right) \geq S_2 \frac{\text{TOL}}{N}, \quad \forall \, i = 1, \ldots, N - 1 \right)$$

$\textbf{then}$ stop.

Here $0 < S_2 < s_2 < s_1 < S_1$ are given constant determined more precisely in [25]. □

Remark 2.9 Here the time steps are chosen to be the same for all components of the solution. Logg [23] uses the more efficient and flexible choice of independent steps for each component of the solution. The error estimate (2.8) would also be applicable to such multi adaptive time steps $\Delta t_{n,i}$ by replacing $\Delta t_n^{p+1} \rho_n$ with $\sum_n \sum_i \Delta t_{n,i}^{p+1} \rho_{n,i}$, where $\rho_{n,i} \equiv \bar{e}_i(t_n) \bar{\Psi}_i(t_n) / \Delta t_{n,i}^{p+1}$. □

Remark 2.10 The discussion on the non convex sets (1.6) in the introduction is inspired by the work [30], which includes an elegant proof of Bakhvalov and Smolyak's result following [4]: assume that $\mathcal{F}$ is a convex symmetric subset of a normed linear function space and that $S : \mathcal{F} \to \mathbb{R}$ is a linear functional with an approximation $S_N(f) = \phi(L_1(f), ..., L_N(f))$, based on N linear functionals $L_k : \mathcal{F} \to \mathbb{R}$, which may depend on $f \in \mathcal{F}$, and a linear or nonlinear mapping $\phi : \mathbb{R}^N \to \mathbb{R}$. Let the worst case error for S_N be defined by

$$\Delta_{\max}(S_N) \equiv \sup_{f \in \mathcal{F}} |S(f) - S_N(f)|,$$

and assume that S_N uses the linear functionals $\{L_k^0\}_{k=1}^N$ for $f \equiv 0 \in \mathcal{F}$. Then there exists $a = (a_1, \ldots, a_N) \in \mathbb{R}^N$ and a linear non-adaptive method

$$S_N^*(f) \equiv \sum_{k=1}^N a_k L_k^0(f),$$

such that

$$\Delta_{\max}(S_N^*) \leq \Delta_{\max}(S_N).$$

As an illustrative example consider the computation of integrals $S(f) = \int_0^1 f(t)dt$ with the approximation $S_N(f)$ based on N point values $L_k(f) = f(t_k)$ at mesh points t_k, $k = 1, \ldots, N$. The method is adaptive if t_k depends on f and non adaptive otherwise. $\square$

Remark 2.11 Let us consider integration of a function by the first order accurate Euler method. Then the integration error is the same as the L^1 approximation error by piecewise constant functions. De-Vore points out in [10] that this L^1 approximation error of a function f, with N non-adaptive steps, is $\mathcal{O}(N^{-\alpha})$ provided f belongs to the Besov space $B_\infty^\alpha(L^1[0,T])$, $\alpha \leq 1$. With N adaptive steps the error is $\mathcal{O}(N^{-\alpha'})$ provided f belongs to the Besov space $B_\infty^{\alpha'}(L^\gamma[0,T])$, $\alpha' \leq 1$, for some $\gamma > (\alpha' + 1)^{-1}$. For $\alpha' \to 1-$, this explains that adaptive integration is better when $\|f'\|_{L^{\frac{1}{2}}(0,1)} << \|f'\|_{L^1(0,1)}$, cf. (2.35), (2.36). $\square$

3 Numerical Experiments

This section presents numerical experiments with the adaptive algorithm `MSTZ` in Section 2, using the `MATLAB` version 5.3 software package, cf. [19]. To study its performance, we choose the Lorenz problem and a problem with a singularity and we compare the results to the adaptive algorithm `ODE45` in MATLAB and to a constant step size algorithm, denoted `Uniform`. In particular, we study the quality of the error estimate, by comparing the ratio between the exact error and the approximate error in (2.16), defined by

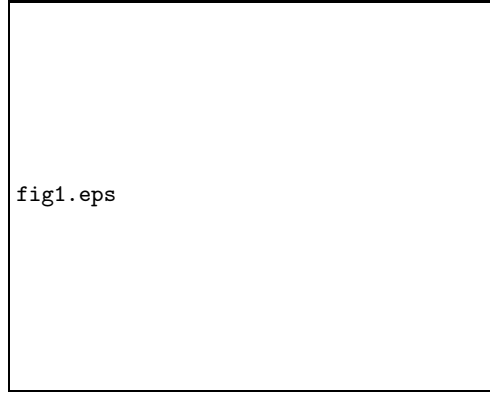$$\Gamma \equiv \frac{|\mathrm{E_T}|}{|g(X(T)) - g(\overline{X}(T))|}. \tag{3.1}$$

Fig. 3.1. Example 3.1: Approximate X_1-component from `MSTZ` with TOL $=$ 10^{-1}.

We also compare the final number of time steps, $N_f \equiv N[J]$, and the total number of time steps, defined by

$$N^{\mathrm{tot}} \equiv \sum_{k=1}^{J} N[k], \qquad (3.2)$$

where J is the total number of refinement levels of `MSTZ`. Similarly, the total number of steps, N^{tot}, for `Uniform` is the total sum of the numbers of steps increasing a factor of 2 on each level, from the same initial number of step $N[1]$, until `Error` $\leq$ TOL. Finally, the N^{tot} for `ODE45` is the total sum of the numbers of steps with `TOL` decreasing a factor of 10, from the same TOL as `MSTZ`, until `Error` $\leq$ TOL.

We initialize `MSTZ` by setting $s_1 = 2$ in (2.19), $S_1 = 2Ms_1$ by (2.23), and $M = 2$.

Example 3.1 Consider the well-known Lorenz system, which is the three dimensional system of ordinary differential equations,

$$\begin{aligned} a_1(t, x) &= -\sigma x_1 + \sigma x_2, \\ a_2(t, x) &= rx_1 - x_2 - x_1 x_3, \quad 0 \leq t \leq T, \quad x \in \mathbb{R}^3, \qquad (3.3) \\ a_3(t, x) &= x_1 x_2 - bx_3 \end{aligned}$$

where σ , r and b are given positive constants.

The Lorenz system was introduced to show the limitation of large time prediction for a simplified model of weather forecast, see [24] and Figure 3.1. In our experiments, the coefficient values are $\sigma = 10$, $b = 8/3$ and $r = 28$ and the initial value is $X(0) = (1, 0, 0)$, cf. [7] and [16]. The computed function is $g(x) = x_1$, i.e., we study the
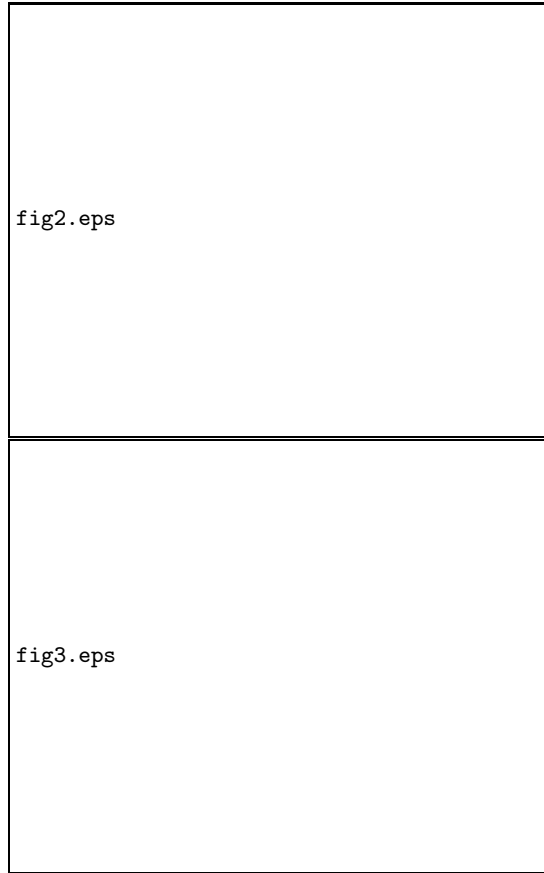
Fig. 3.2. Example 3.1: Comparison of the mesh functions of MSTZ and ODE45. The minimum value of Δt of MSTZ is 0.0016 which is 22 times larger than the minimum of ODE45.

global error $|X_1(T) - \overline{X}_1(T)|$ at the final time $T = 30$. A reference computation with a Fortran implementation of MSTZ in quadruple precision gives the approximate value $X_1(30) \simeq -3.892637 \equiv g_1$, with TOL $= 10^{-7}$.

ODE45 is based on an explicit Runge-Kutta (5,4) formula, the Dormand-Prince pair, see [11]. In order to compare with ODE45, the program MSTZ also uses the same 5-th order explicit Runge-Kutta method to compute $\overline{X}(t_{n+1})$, $\overline{\Psi}(t_n)$ and $\overline{\overline{X}}(t_{n+1})$. The approximate local "exact" solution $\overline{\overline{X}}$ is computed with the half mesh size, i.e., $\gamma = 2^5/(2^5 - 1)$ in (2.7). The program Uniform uses a constant step size, $\Delta t = constant$, based on the same 5-th order explicit Runge-Kutta method. Table 1 and Figure 3.2 show that the algorithm MSTZ

	Tolerance	Error	N_f	N^{tot}
MSTZ	TOL $= 10^{-1}$	0.01	6000	20000
	TOL $= 10^{-2}$	0.003	9000	34000
Uniform	TOL $= 10^{-1}$	0.06	10000	19000
	TOL $= 10^{-2}$	0.002	19000	38000
ODE45	TOL$= 10^{-10}$	0.04	34000	92000
	TOL$= 10^{-11}$	0.004	53000	144000

Table 1. Example 3.1: Comparisons of the final number of steps, N_f, and the total number of steps, N^{tot}, with the global error, Error $\equiv |g_1 - g(\overline{X}(T))|$, using a 5-th order explicit Runge-Kutta method with adaptive steps for MSTZ or ODE45 and uniform steps for Uniform.

achieves higher accuracy with around half the final number of time steps compared to Uniform and with one fifth of the final number of steps compared to ODE45. The total number of steps for ODE45 is around four times the total number of steps of MSTZ, including all 7 and 8 refinement levels with $N[1] = 300$. Table 1 also shows the difference between the tolerance parameter TOL in MSTZ and the parameter TOL in ODE45: TOL measures the global error, while TOL measures a combination of the relative local error and absolute local error.

The function c defined by (2.22) is used in the assumptions of all theorems. Figure 3.3 motives our effort to not only base the assumptions on $\max c^{-1}$, which can be very large. The assumptions based on the c^{-1} for the step with maximal error indicator and the weighted average $\|\frac{\bar{\rho}}{c}\|_{L^{\frac{1}{p+1}}} / \|\bar{\rho}\|_{L^{\frac{1}{p+1}}}$ give better estimates, since these measures of c^{-1} are close to 1.

Example 3.2 Consider (1.2) with

$$a(t, x) = \frac{x}{\sqrt{|t - \omega|}}, \quad 0 \le t \le T, \quad x \in \mathbb{R} \tag{3.4}$$

where $\omega \in [0, T]$ is a constant, and let $g(x) = x$ and $X(0) = e^{-2\sqrt{\omega}}$. The exact solution is then $X(t) = e^{sign(t-\omega)\, 2\sqrt{|t-\omega|}}$.

The function a in (3.2) has a singularity at $t = \omega$. The corresponding error density (2.10), $\rho_p \equiv \rho$ for a p-th order method, satisfies $\|\rho_p\|_{L^1} = \infty$ for $p \ge 1$ and interpolation between the first order ρ_1 and the zero order $\rho_0 \equiv a$ shows $\|\rho_p\|_{L^1} < \infty$ for $p < 1/2$, so that by (2.36) the number of uniform steps becomes $N^{\mathrm{uni}} \sim \mathrm{TOL}^{-2}$. In contrast, the convergence rate for adaptive approximation remains by (2.35) optimal $N = \mathcal{O}(\mathrm{TOL}^{-1/p})$, since $\|\rho_p\|_{L^{\frac{1}{p+1}}} < \infty$ for $p > 0$.
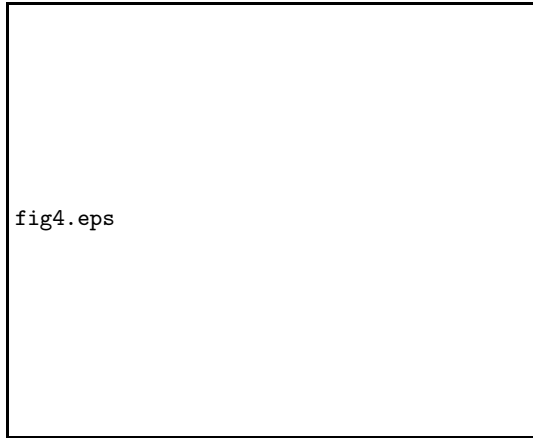
```
fig4.eps
```

Fig. 3.3. Example 3.1: Comparison between $\max c^{-1}$, the c^{-1} for the step with maximal error indicator and the weighted average $\|\frac{\bar{\rho}}{c}\|_{L^{\frac{1}{p+1}}} / \|\bar{\rho}\|_{L^{\frac{1}{p+1}}}$. We see that the last two are close to 1, while $\max c^{-1}$ is very large, which motivates the assumptions in the theorems.

```
fig5.eps
```

```
fig6.eps
```

Fig. 3.4. Example 3.2: Approximate solution (up) and mesh function (down) of MSTZ using a 5-th order explicit Runge-Kutta method with TOL $= 10^{-4}$ and $\omega = 5/3$.

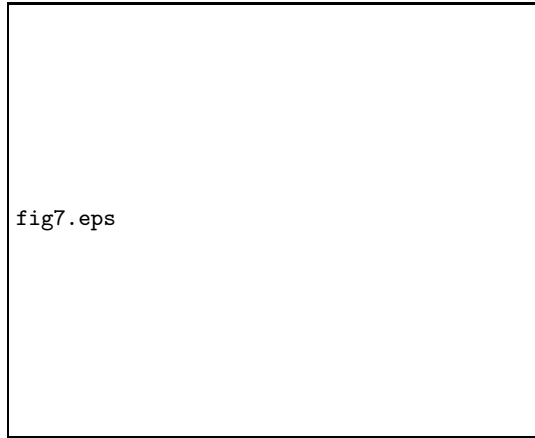**Fig. 3.5.** Example 3.2: The ratio of the approximate and exact error, Γ, converges to 1 as TOL $\rightarrow$ 0+ with the stopping condition (2.20), while Γ does not converge to 1 with the alternative stopping condition $|\mathtt{E_T}| \leq$ TOL. In general the condition $|\mathtt{E_T}| \leq$ TOL stops the program earlier and for some examples the error ratio Γ is not as accurate as with the condition (2.20).

Consider the case $\omega = 5/3$ with $T = 4$, $N[1] = 2^5$ and TOL $= 10^{-1}, 10^{-4}$. Table 2 and Figure 3.4 show that `MSTZ` and `ODE45` are much more efficient than `Uniform`, as expected.

	Tolerance	Error	N_f	N^{tot}
`MSTZ`	TOL $= 10^{-1}$	0.02	50	820
	TOL $= 10^{-4}$	2.6×10^{-5}	130	3880
`Uniform`	TOL $= 10^{-1}$	0.06	130000	260000
		0.015	2100000	4200000
`ODE45`	TOL$= 10^{-5}$	0.02	210	480
	TOL$= 10^{-8}$	4.8×10^{-6}	710	1800

Table 2. Example 3.2: Comparisons of the final number of steps, N_f, and the total number of steps, N^{tot}, with the global error, $\mathtt{Error} \equiv |X(T) - \overline{X}(T)|$, using a 5-th order explicit Runge-Kutta method with adaptive steps for `MSTZ` or `ODE45` and uniform steps for `Uniform`. Adaptive approximation is more efficient for this singularity.

Figure 3.5 compares stopping by (2.20) and the alternative $|\mathtt{E_T}| \leq$ TOL. In general, the condition $|\mathtt{E_T}| \leq$ TOL stops the program earlier than with the stopping condition (2.20) and sometimes this yields a less accurate error estimate as in Figure 3.5.
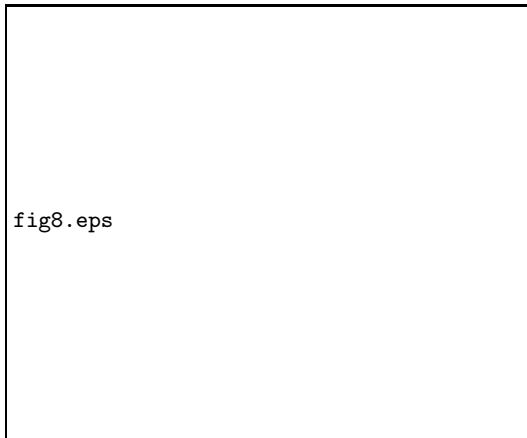
Fig. 3.6. Example 3.1: The error indicators and the first component of the weights and the local errors from MSTZ with TOL $= 10^{-1}$. The other two components have a similar behavior. Note that the local errors and the weights oscillate, but their product does not give a significant error cancellation, see Table 3.

Remark 3.3 If a time node, say t_{m+1}, hits the singularity at $t = \omega$, the approximation $\overline{X}(t_{m+1})$ becomes an *infinite* number. A remedy for this is to change the time steps, i.e $t_m := t_m + \alpha$ and $t_{m+1} := t_{m+1} + \beta$ where α and β are sufficiently small numbers, e.g. $\Delta t_m/M$, and then recompute $\overline{X}(t_m)$ and $\overline{X}(t_{m+1})$. Using this technique, we solve Example 3.2 for $w = 1$ and $T = 4$ and we get $|g(X(T)) - g(\overline{X}(T))| = 1.3065 \times 10^{-4}$ with 113 final time steps and 2567 total time steps using a 5-th order explicit Runge-Kutta method and TOL $= 10^{-3}$, $N[1] = 40$.   □

	Lorenz (Example 3.1)		Singularity (Example 3.2)	
Tolerance	10^{-1}	10^{-2}	10^{-1}	10^{-4}
Γ	0.991	0.997	1.325	2.31
$\overline{\Gamma}$	1.707	1.220	1.325	2.66

Table 3. Example 3.1 and 3.2: Comparisons of the ratio, Γ and $\overline{\Gamma}$, between the exact error and the approximate error using error density, $\bar{\rho}$ and $|\bar{\rho}|$ respectively, for MSTZ.

Remark 3.4 The algorithm MSTZ does not use the sign of the error density. Therefore the computational error, $\mathrm{E_T} \equiv \sum_{i=1}^{N} \bar{\rho}_i \Delta t_i^{p+1}$ in (2.16) could be much smaller than TOL $\geq (1/S_1) \sum_{i=1}^{N} |\bar{\rho}_i| \Delta t_i^{p+1}$,

when it stops. Table 3 compares the quantities Γ in (3.1) and $\overline{\Gamma} \equiv |\sum_{i=1}^{N} |\bar{\rho}_i| \Delta t_i^{p+1}|/|g(X(T)) - g(\overline{X}(T))|$ for the Examples 3.1 and 3.2. Figure 3.6 shows the error indicators and the first component of the weights and the local errors in Example 3.1. We observe that the cancellation of the error in $\sum_{i=1}^{N} \bar{\rho}_i \Delta t_i^{p+1}$ only yields a factor of two reduction compared to $\sum_{i=1}^{N} |\bar{\rho}_i| \Delta t_i^{p+1}$. Therefore the time steps determined by the error bound $\sum_{i=1}^{N} |\bar{\rho}_i| \Delta t_i^{p+1}$, ignoring the sign and the cancellation of the error, are also almost optimal taking cancellation into account, for these two examples. The cancellation of the error is very important for Itô stochastic differential equations. In [33] we use properties of Brownian motion to derive an error density which takes cancellation into account.   □

References

1. M. Ainsworth and J. T. Oden, A posteriori error estimation in finite element analysis, Comput. Methods Appl. Mech. Engrg., **142** (1997), 1-88.
2. I. Babuška, A. Miller and M. Vogelius, Adaptive methods and error estimation for elliptic problems of structural mechanics, in *Adaptive computational methods for partial differential equations* (SIAM, Philadelphia, Pa., 1983) 57-73.
3. I. Babuška, and M. Vogelius, Feedback and adaptive finite element solution of one-dimensional boundary value problems, Numer. Math. **44** (1984), no. 1, 75-102.
4. N.S. Bakhvalov, On the optimality of linear methods for operator approximation in convex classes of functions, USSR Comput. Math. and Math. Phys., **11** (1971), 244-249.
5. R. Becker and R. Rannacher, A feed-back approach to error control in finite element methods: basic analysis and examples, East-West J. Numer. Math., **4** (1996), no. 4, 237-264.
6. R. Becker and R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods Acta Numerica, (2001), 1-102.
7. K. Böttcher and R. Rannacher, Adaptive error control in solving ordinary differential equations by the discontinuous Galerkin method, preprint, (1996).
8. A. Cohen, W. Dahmen and R. DeVore, Adaptive wavelet methods for elliptic operator equations: convergence rates, Math. Comp., **70** (2001), no. 233, 25-75.
9. G. Dahlquist and Å. Björk, *Numerical Methods*, (Prentice-Hall, 1974).

10. R. A. DeVore, Nonlinear approximation, Acta Numerica, (1998), 51-150.

11. J.R. Dormand and P.J. Prince, A family of embedded Runge-Kutta formulae, J. Comput. Appl. Math., **6** (1980), no. 1, 19–26.

12. W. Dörfler, A convergent adaptive algorithm for Poisson's equation, SIAM J. Numer. Anal. **33** (1996), no. 3, 1106-1124.

13. K. Eriksson, D. Estep, P. Hansbo and C. Johnson, Introduction to adaptive methods for differential equations, Acta Numerica, (1995), 105-158.

14. D. Estep, A posteriori error bounds and global error control for approximation of ordinary differential equations, SIAM J. Numer. Anal., **32** (1995), 1-48.

15. D. Estep, D. Hodges and M. Warner, Computational error estimates and adaptive error control for a finite element solution of launch vehicle trajectrory problems, SIAM J. SCI. COMPUT., **21** (2000), 1609-1631.

16. D. Estep and C. Johnson, The pointwise computability of the Lorenz system, Math. Models Methods Appl. Sci., **8** (1998), 1277-1305.

17. F. Gao, Probabilistic analysis of numerical integration algorithms, J. Complexity **7** (1991), no. 1, 58-69.

18. E. Harrier, S.P. Norsett and G. Wanner, *Solving Ordinary Differential Equations I*, (Springer-Verlag, 1993).

19. D.J. Higham and N.J. Higham , *Matlab Guide*, (Society for Industrial and Applied Mathematics, 2000).

20. C. Johnson, Error estimates and adaptive time-step control for a class of one-step methods for stiff ordinary differential equations, SIAM J. Numer. Anal., **25** (1988), 908-926.

21. C. Johnson and A. Szepessy, Adaptive finite element methods for conservation laws based on a posteriori error estimates, Comm. Pure Appl. Math., **48** (1995), 199-234.

22. H. Lamba and A. M. Stuart, Convergence results for the MATLAB ODE23 routine, BIT, **38** (1998), No. 4, 751-780.

23. A. Logg, *Multi-adaptive Galerkin methods for ODEs,* (Licentiate thesis, ISSN 0347-2809, Chalmers University of Technology, 2001), http://www.md.chalmers.se/Centres/Phi/preprints/index.html.

24. E. N. Lorenz, Deterministic non-periodic flows, J. Atmos. Sci., **20** (1963), 130–141.

25. K.-S. Moon, *Convergence rates of adaptive algorithms for deterministic and stochastic differential equations*, (Licentiate thesis, ISBN 91-7283-196-0, Royal Institute of Technology, 2001), http://www.nada.kth.se/∼moon/paper.html.

26. K.-S. Moon, E. von Schwerin, A. Szepessy and R. Tempone, Convergence rates for adaptive finite element approximation of partial differential equations, work in progress.

27. K.-S. Moon, A. Szepessy, R. Tempone and G.E. Zouraris, Hyperbolic differential equations and adaptive numerics, in *Theory and numerics of differential equations* (Eds. J.F. Blowey, J.P. Coleman and A.W. Craig, Durham 2000, Springer Verlag, 2001).

28. K.-S. Moon, A. Szepessy, R. Tempone and G.E. Zouraris, A variational principle for adaptive approximation of ordinary differential equations, Numer. Math. (in press).

29. P. Morin, R. Nochetto and K.G. Siebert, Data oscillation and convergence of adaptive FEM, SIAM J. Numer. Anal., **38** (2000), 466-488.

30. E. Novak, On the power of adaption, J. Complexity, **12** (1996), 199-237.

31. A.M. Stuart, Probabilistic and detereministic convergence proofs for software for initial value problems, Numerical Algorithms, **14** (1997), 227-260.
32. A. Szepessy and R. Tempone, Optimal control with multigrid and adaptivity, Fifth World Congress on Computational Mechanics, http://wccm.tuwien.ac.at.
33. A. Szepessy, R. Tempone and G. E. Zouraris, Adaptive weak approximation of stochastic differential equations, Comm. Pure Appl. Math., **54** (2001), 1169-1214.
34. G. Söderlind, Automatic control and adaptive time-stepping ANODE01 Proceedings, Numerical Algorithms.
35. J.F. Traub and A.G. Werschulz, Complexity and Information, Cambridge University Press, Cambridge, 1998.
36. T. Utumi, R. Takaki and T. Kawai, Optimal time step control for the numerical solution of ordinary differential equations, SIAM J. Numer. Anal., **33** (1996), 1644-1653.
37. A. G. Werschulz, The Computational Complexity of Differential and Integral Equations, An Information-Based Approach, Oxford Mathematical Monographs. Oxford Science Publications. The Clarendon Press, Oxford University Press, New York, 1991.