Adaptive mesh selection asymptotically guarantees a prescribed local error for systems of initial value problems [1]

Bolesław Kacewicz [2]

arXiv:1704.01939v3 [math.NA] 22 Dec 2017

Abstract

We study potential advantages of adaptive mesh point selection for the solution of systems of initial value problems. For an optimal order discretization method, we propose an algorithm for successive selection of the mesh points, which only requires evaluations of the right-hand side function. The selection (asymptotically) guarantees that the maximum local error of the method does not exceed a prescribed level. The usage of the algorithm is not restricted to the chosen method; it can also be applied with any method from a general class. We provide a rigorous analysis of the cost of the proposed algorithm. It is shown that the cost is almost minimal, up to absolute constants, among all mesh selection algorithms. For illustration, we specify the advantage of the adaptive mesh over the uniform one. Efficiency of the adaptive algorithm results from automatic adjustment of the successive mesh points to the local behavior of the solution. Some numerical results illustrating theoretical findings are reported.

Mathematics Subject Classification: 65L05, 65L50, 65L70

---

[1] This research was partly supported by the Polish NCN grant - decision No. DEC-2017/25/B/ST1/00945 and by the Polish Ministry of Science and Higher Education

[2] *AGH University of Science and Technology, Faculty of Applied Mathematics,*
Al. Mickiewicza 30, paw. A3/A4, III p., pok. 301,
30-059 Cracow, Poland
E-mail B. Kacewicz: kacewicz@agh.edu.pl

1 Introduction

We deal with the solution of systems of initial value problems (IVPs)

$$z'(t) = f(t, z(t)), \quad t \in [a, b], \qquad z(a) = \eta, \tag{1}$$

where $a < b$, $\eta \in \mathbf{R}^d$ and $f : [a, b] \times \mathbf{R}^d \to \mathbf{R}^d$ is a C^r function. We study how much adaptive mesh points improve efficiency of algorithms for solving (1). For many years, adaption has been a standard tool in numerical packages. Well known examples include the package QUADPACK [8] for numerical integration, or, among other solvers, the DIFSUB procedure by C.W. Gear or the library ODEPACK by A. Hindmarsh for IVPs. Many authors have reported superiority of adaption over nonadaption for (1), based on numerical results for a number of computational examples. One can cite as an example papers such as [3], [6] or [7]. Practical efficiency gives us considerable knowledge about the power of adaption. Conclusions are however not complete; an analysis of theoretical aspects is missing in many cases of step size control strategies. For instance, most often, step size control devices are not supported by cost analysis. The use of variable step size not only improves the efficiency of methods for solving regular problems (1), but it also allows us to manage singularities. Considerable progress has been made in rigorous analysis of adaption in that case, see e.g. [5], [10]. Adaption allows us in many cases to maintain for singular problems the order of convergence known from the regular case.

Advantages of adaption for regular problems are of different type. Integration of scalar C^4 functions by the Simpson rule have been recently studied in [9]. It is shown in [9] that adaption does not improve the order of convergence, but it can reduce the asymptotic constant of the method. In the similar spirit, adaption has been considered for univariate approximation and minimization in [1]. For scalar autonomous problems (1), adaptive mesh selection has been recently studied in [4]. An adaptive strategy has been proposed and the cost analyzed, based on specific properties of scalar autonomous equations. The particular technique used does not allow us to extend the results from [4] to systems of IVPs.

In the present paper, we consider general systems (1), with a C^r right-hand side function f. Our contribution can be summarized as follows:

- For a maximal order method for solving the system (1), we propose a new mesh selection algorithm that guarantees the local error of the method at a precribed level ε, for sufficiently small $\varepsilon \in (0, 1)$. At each time step, we select the step size and compute an approximation to the solution, which requires two runs of the basic method. Information about the function f only consists of function evaluations. We show that the mesh selection algorithm can be applied to a general class of methods for solving (1).

- We rigorously analyze the cost of the method equipped with the mesh selection procedure. We show that the cost is minimal among all mesh selection strategies, provided that we accept some absolute constants.
- We specify in a quantitative way an advantage of the adaptive mesh over the uniform one.

The adaptive mesh selection for systems (1) does not improve the speed of convergence of algorithms. A potential gain of adaption lies in reducing a coefficient in the error bound.
The paper is organized as follows. We formulate the problem in Section 2. An algorithm ϕ^* with maximal order is defined in Section 3, and the error analysis is given in Theorem 1. In Section 4, we discuss upper bounds on the local error of ϕ^*, in particular, we show a constructive upper bound in Theorem 2. Section 5 contains the main algorithm ADAPT-MESH which combines ϕ^* with a new mesh selection algorithm. A generalization of the algorithm ADAPT-MESH is given in Section 6. Section 7 contains a cost analysis of the algorithm ADAPT-MESH compared to other algorithms, see Theorem 3. Possible advantage of the adaptive mesh over the uniform one is discussed in detail. Finally, some results of numerical experiments are reported in Section 8.

2 Problem formulation

We consider problem (1) with a continuous function $f : [a, b] \times \mathbf{R}^d \to \mathbf{R}^d$ such that for some $L \geq 0$

$$\|f(t, y) - f(t, \bar{y})\| \leq L\|y - \bar{y}\| \quad \text{for } t \in [a, b], \ \ y, \bar{y} \in \mathbf{R}^d. \tag{2}$$

Here and in what follows $\|\cdot\|$ denotes the maximum norm in $\mathbf{R}^d$. It follows that there exists a unique solution z of (1) defined on $[a, b]$. Let $r \geq 1$. We assume that f is a regular function in a subset of its domain, $f \in C^r([a, b] \times D)$, where

$$D = \{y \in \mathbf{R}^d : \ \|y\| \leq \sup_{t \in [a,b]} \|z(t)\| + 1\}. \tag{3}$$

The class of functions f satisfying the above assumptions will be denoted by F_r.
Let $m \in \mathbf{N}$. We wish to compute an approximation to the solution z in $[a, b]$. For $m + 1$ mesh points $a = x_{0,m} < x_{1,m} < \ldots < x_{m,m} = b$, we do it by computing approximations l_i to z in the subintervals $[x_{i,m}, x_{i+1,m}]$, $i = 0, 1, \ldots, m - 1$.

Let $\ell(m)$ be any nonincreasing sequence convergent to 0 as $m \to \infty$. We consider for any f a class of partitions of $[a, b]$ defined as follows. We assume that there exist $K = K(f, a, b, \eta)$ and $k_0 = k_0(f, a, b, \eta)$ such that for all $m \geq k_0$ and any partition it holds

$$\max_{0 \leq i \leq m-1} (x_{i+1,m} - x_{i,m}) \leq K\, \ell(m). \tag{4}$$

Thus, the partitions under consideration are uniformly normal. Note that we always have $\max\limits_{0 \leq i \leq m-1} (x_{i+1,m} - x_{i,m}) \geq (b - a)/m$ for any $m \geq 1$. Thus, the condition (4) implies that $\ell(m)$ cannot go to zero faster than $1/m$. Note that the convergence of $\ell(m)$ to zero can be arbitrarily slow, and the constant K can be arbitrarily large. We shall omit in the sequel the second subscript m, remembering that the choice of points x_i can be different for varying m.
For a given $y_i \in \mathbf{R}^d$, we denote by z_i the solution of a local problem

$$z_i'(t) = f(t, z_i(t)), \quad t \in [x_i, x_{i+1}], \qquad z_i(x_i) = y_i. \tag{5}$$

If l_i is an approximation to z_i given by a certain method, then local errors of the method are given by

$$\sup_{t \in [x_i, x_{i+1}]} \|z_i(t) - l_i(t)\|, \qquad i = 0, 1, \ldots, m-1.$$

Our aim is to select possibly small m and mesh points $\{x_i\}_{i=0}^m$ such that the local errors remain at a precribed level $\varepsilon > 0$.

3 The basic method ϕ^* and its error

The basic method makes use of the approximate Picard iteration, an idea that turned out useful in several contexts, see e.g. [2], [5]. Let $m \in \mathbf{N}$ and $x_0 = a < x_1 < \ldots < x_m = b$ be mesh points satisfying (4). Let $y_0 = \eta$. For a given y_i, we define approximations $l_{i,j}$ in $[x_i, x_{i+1}]$ as follows.
We set $l_{i,0}(t) \equiv y_i$. Let $l_{i,j}$ be given. Denote by $t_0, t_1, \ldots, t_{r-1}$ the equidistant nodes in $[x_i, x_{i+1}]$, with $t_0 = x_i$ for $r = 1$ and $t_0 = x_i$, $t_{r-1} = x_{i+1}$ for $r \geq 2$. (The points t_k depend on i; we shall omit this index in the notation.) We define $q_{i,j}$ to be the Lagrange interpolation polynomial of degree at most $r - 1$ for the function $g_{i,j}(t) = f(t, l_{i,j}(t))$ based on the nodes $t_0, t_1, \ldots, t_{r-1}$,

$$q_{i,j}(t) = \sum_{k=0}^{r-1} g_{i,j}(t_k) \prod_{p=0, p\neq k}^{r-1} \frac{t - t_p}{t_k - t_p}, \qquad t \in [x_i, x_{i+1}], \tag{6}$$

where $\prod_{p=0, p\neq k}^{0} = 1$. An approximation $l_{i,j+1}$ is given by

$$l_{i,j+1}(t) = y_i + \int_{x_i}^{t} q_{i,j}(\xi)\, d\xi, \quad t \in [x_i, x_{i+1}]. \tag{7}$$

The final approximation in $[x_i, x_{i+1}]$ is given by $l_{i,r+1}$. To complete the definition, we set $y_{i+1} = l_{i,r+1}(x_{i+1})$.
For $t \in [a, b]$ we define a continuous approximation to z by

$$l_{r+1}(t) = l_{i,r+1}(t), \quad t \in [x_i, x_{i+1}]. \tag{8}$$

The transformation that assignes to f the approximation l_{r+1} will be denoted by ϕ^*, $\phi^*(f)(t) = l_{r+1}(t)$, $t \in [a, b]$.
The following theorem provides error analysis of the method ϕ^*. The proof follows usual lines of the analysis of approximate Picard iteration, it is however focused on our specific requirements. We shall need in the next sections the error bound (9) for a non-uniform mesh, as well as specific local error bounds derived in the body of the proof. Let $h_i = x_{i+1} - x_i$.

Theorem 1 *Let $f \in F_r$. There exists m_0 such that for all $m \geq m_0$ and any $\{x_i\}_{i=0}^m$ satisfying (4), the global error of ϕ^* at f satisfies*

$$\sup_{t \in [a,b]} \|z(t) - l_{r+1}(t)\| \leq M \max_{0 \leq j \leq m-1} h_j^r, \tag{9}$$

where $M = \exp(L(b-a))(b-a)\,(2D_r/r! + 1/2)$, and the number D_r, only dependent on f, r, a, b, is defined below before the inequality (17).

Proof Let $M_0 = 0$ and $M_{i+1} = \exp(Lh_i)M_i + (2D_r/r! + 1/2)\, h_i$, $i = 0, 1, \ldots, m-1$. One can check that $M_i \leq M$, $i = 0, 1, \ldots, m$. We shall show by induction on i that

$$\sup_{t \in [a,x_i]} \|z(t) - l_{r+1}(t)\| \leq M_i \max_{0 \leq j \leq i-1} h_j^r, \tag{10}$$

where $\max_{0 \leq j \leq -1} = 1$.
For $i = 0$, (10) holds true. Let (10) hold for some i. Consider the interval $[x_i, x_{i+1}]$. We have that

$$\|z(t) - z_i(t)\| \leq \exp(Lh_i)\, \|z(x_i) - y_i\|, \quad t \in [x_i, x_{i+1}]. \tag{11}$$

We shall now study the local error in $[x_i, x_{i+1}]$ given by $e_{i,j} = \sup_{t \in [x_i, x_{i+1}]} \|z_i(t) - l_{i,j}(t)\|$.
Denoting $H_i(t) = f(t, z_i(t))$ $(= z_i'(t))$, we let $\bar{q}_i$ be the Lagrange interpolation polynomial for

H_i,

$$\bar{q}_i(t) = \sum_{k=0}^{r-1} f(t_k, z_i(t_k))) \prod_{p=0,p\neq k}^{r-1} \frac{t-t_p}{t_k-t_p}, \quad t \in [x_i, x_{i+1}]. \tag{12}$$

Since

$$z_i(t) = y_i + \int_{x_i}^{t} f(\xi, z_i(\xi))\, d\xi,$$

we have that

$$\|z_i(t) - l_{i,j+1}(t)\| \leq \int_{x_i}^{t} \|f(\xi, z_i(\xi)) - \bar{q}_i(\xi)\|\, d\xi + \int_{x_i}^{t} \|\bar{q}_i(\xi) - q_{i,j}(\xi)\|\, d\xi. \tag{13}$$

By the Lagrange interpolation error formula applied component by component, we have that

$$\|f(\xi, z_i(\xi)) - \bar{q}_i(\xi)\| \leq \frac{1}{r!} \sup_{\alpha\in[x_i,x_{i+1}]} \|H_i^{(r)}(\alpha)\| \prod_{k=0}^{r-1} |\xi - t_k|, \quad \xi \in [x_i, x_{i+1}].$$

Furthermore,

$$\|\bar{q}_i(\xi) - q_{i,j}(\xi)\| \leq \sum_{k=0}^{r-1} \|f(t_k, z_i(t_k)) - f(t_k, l_{i,j}(t_k))\| \prod_{p=0,p\neq k}^{r-1} \left|\frac{\xi - t_p}{t_k - t_p}\right|,$$

which yields that

$$\|\bar{q}_i(\xi) - q_{i,j}(\xi)\| \leq L\hat{C}_r \sup_{t\in[x_i,x_{i+1}]} \|z_i(t) - l_{i,j}(t)\|, \quad \xi \in [x_i, x_{i+1}],$$

where $\hat{C}_r$ only depends on r. From (13) we get for $j = 0, 1, \ldots$

$$e_{i,j+1} \leq \frac{1}{r!} \sup_{\alpha\in[x_i,x_{i+1}]} \|H_i^{(r)}(\alpha)\|\, h_i^{r+1} + h_i L\hat{C}_r e_{i,j}. \tag{14}$$

By solving (14) we get for $j = 0, 1, \ldots$

$$e_{i,j} \leq \frac{1}{r!} \sup_{\alpha\in[x_i,x_{i+1}]} \|H_i^{(r)}(\alpha)\|\, h_i^{r+1}\, \frac{1 - (h_i L\hat{C}_r)^j}{1 - h_i L\hat{C}_r} + (h_i L\hat{C}_r)^j e_{i,0}.$$

Since

$$e_{i,0} \leq h_i \sup_{\alpha\in[x_i,x_{i+1}]} \|f(\alpha, z_i(\alpha))\|,$$

for m sufficiently large (such that $h_i L\hat{C}_r \leq 1/2$) we have that

$$e_{i,j} \leq \frac{2}{r!} \sup_{\alpha\in[x_i,x_{i+1}]} \|H_i^{(r)}(\alpha)\|\, h_i^{r+1} + h_i^{j+1} (L\hat{C}_r)^j \sup_{\alpha\in[x_i,x_{i+1}]} \|H_i(\alpha)\|. \tag{15}$$

Note that $H_i^{(r)}(\alpha)$ and $H_i(\alpha)$ can be expressed in terms of partial derivatives of the function f of order $0, 1, \ldots, r$, evaluated at $(\alpha, z_i(\alpha))$. Due to (11) and the inductive assumption, for sufficiently large m we have that $\|z(\alpha) - z_i(\alpha)\| \leq 1$ for $\alpha \in [x_i, x_{i+1}]$. Hence,

$$(\alpha, z_i(\alpha)) \in [a, b] \times D, \tag{16}$$

where the set D is given in (3). This yields that $\sup_{\alpha \in [x_i, x_{i+1}]} \|H_i^{(r)}(\alpha)\|$ and $\sup_{\alpha \in [x_i, x_{i+1}]} \|H_i(\alpha)\|$ are bounded from above independently of i and m by some numbers D_r and D_0, respectively. For the final approximation $l_{i,r+1}$ we have from (15) that

$$e_{i,r+1} \leq \frac{2}{r!} D_r h_i^{r+1} + h_i^{r+2} (L\hat{C}_r)^{r+1} D_0, \tag{17}$$

which yields for sufficiently large m that

$$e_{i,r+1} \leq \left(\frac{2}{r!} D_r + \frac{1}{2} \right) h_i^{r+1}. \tag{18}$$

For the final global approximation l_{r+1}, we have for $t \in [x_i, x_{i+1}]$

$$\|z(t) - l_{r+1}(t)\| \leq \|z(t) - z_i(t)\| + \|z_i(t) - l_{i,r+1}(t)\| \leq \exp(Lh_i)\|z(x_i) - y_i\| + e_{i,r+1}. \tag{19}$$

By the inductive assumption, we get that

$$\|z(t) - l_{r+1}(t)\| \leq \exp(Lh_i) M_i \max_{0 \leq j \leq i-1} h_j^r + \left(\frac{2}{r!} D_r + \frac{1}{2} \right) h_i^{r+1}, \quad t \in [x_i, x_{i+1}]. \tag{20}$$

Hence,

$$\sup_{t \in [a, x_{i+1}]} \|z(t) - l_{r+1}(t)\| \leq M_{i+1} \max_{0 \leq j \leq i} h_j^r, \tag{21}$$

where $M_{i+1} = \exp(Lh_i) M_i + (2D_r/r! + 1/2)\, h_i$. The induction is finished.
To complete the proof we recall that $M_i \leq M$ for $i = 0, 1, \ldots, m$, where M is given in the statement of the theorem. ■

Given m and a mesh $\{x_i\}_{i=0}^m$, the method ϕ^* describes the construction of the approximations $l_{i,r+1}$ in $[x_i, x_{i+1}]$ for $i = 0, 1, \ldots, m-1$. Theorem 1 provides the bound on the global error of ϕ^*. A selection of the mesh $\{x_i\}_{i=0}^m$ still remains an open question; we will study this issue in the next sections.

4 Local error bounds for ϕ^*

We now extract from the proof of Theorem 1 bounds on the local error $e_{i,r+1}$ of the method ϕ^*. By (15)

$$e_{i,r+1} \leq \frac{2}{r!} \sup_{\alpha \in [x_i, x_{i+1}]} \|H_i^{(r)}(\alpha)\| \, h_i^{r+1} + h_i^{r+2} (L\hat{C}_r)^{r+1} \sup_{\alpha \in [x_i, x_{i+1}]} \|H_i(\alpha)\|. \tag{22}$$

Let $\beta > 0$. From (22), for sufficiently large m we have

$$e_{i,r+1} \leq 2 \left(\frac{1}{r!} \sup_{\alpha \in [x_i, x_{i+1}]} \|H_i^{(r)}(\alpha)\| + \beta \right) h_i^{r+1}. \tag{23}$$

The function $H_i(\alpha) = f(\alpha, z_i(\alpha)) = z_i'(\alpha)$ is not known. We now show how the term 'sup' above can be (asymptotically) replaced by a known quantity.
We take $\bar{x}_{i+1}$, $x_i < \bar{x}_{i+1} \leq b$. For $\bar{h}_i = \bar{x}_{i+1} - x_i$ we assume that

$$\bar{h}_i \leq \gamma h_i, \tag{24}$$

where $\gamma \geq 1$ is a given number which may depend on f, but is independent of i and m.
Let $\bar{t}_k$, $k = 0, 1, \ldots, r$ be equidistant points from $[x_i, \bar{x}_{i+1}]$, $\bar{t}_0 = x_i$, $\bar{t}_r = \bar{x}_{i+1}$ (we omit the index i). We construct an auxillary approximation $\bar{l}_{i,r+1}$ in the interval $[x_i, \bar{x}_{i+1}]$ in the same way as we did in (6)–(8) in the case of the approximation $l_{i,r+1}$ in the interval $[x_i, x_{i+1}]$, using now as interpolation nodes the points $\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_{r-1} \in [x_i, \bar{x}_{i+1}]$.
Let $\tilde{H}_i(\alpha) = f(\alpha, \bar{l}_{i,r+1}(\alpha))$ and $\tilde{H}_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r]$ be the divided difference, computed component by component, for $\tilde{H}_i$. We shall need the bounds stated in the following two lemmas. Recall that $H_i(t) = f(t, z_i(t))$.

Lemma 1 *Let $f \in F_r$, $\beta > 0$ and $\varphi \in (0,1)$. There exists m_0 such that for any $m \geq m_0$, for any $\{x_i\}_{i=0}^m$ satisfying (4) and $i = 0, 1, \ldots, m-1$ we have*

$$(S_i + \beta)(1 - \varphi) \leq \frac{1}{r!} \sup_{\alpha \in [x_i, x_{i+1}]} \|H_i^{(r)}(\alpha)\| + \beta \leq (S_i + \beta)(1 + \varphi), \tag{25}$$

where $S_i = \|H_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r]\|$.

Proof Let $\bar{\alpha}$ be a point from $[x_i, x_{i+1}]$ for which

$$\sup_{\alpha \in [x_i, x_{i+1}]} \|H_i^{(r)}(\alpha)\| = \|H_i^{(r)}(\bar{\alpha})\|.$$

For the lth component H_i^l of the function H_i, we express the divided difference as

$$H_i^l[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r] = \frac{1}{r!} \left(H_i^l\right)^{(r)} (\hat{\alpha}^l), \tag{26}$$

where $\hat{\alpha}^l$ is some point from $[x_i, \bar{x}_{i+1}]$. For any $l = 1, 2, \ldots, d$ it holds

$$\frac{1}{r!} \left| \left(H_i^l\right)^{(r)} (\bar{\alpha}) \right| + \beta = \left(\frac{1}{r!} \left| \left(H_i^l\right)^{(r)} (\hat{\alpha}^l) \right| + \beta \right) (1 + \kappa_i^l),$$

where

$$\kappa_i^l = \frac{\left| \left(H_i^l\right)^{(r)} (\bar{\alpha}) \right| /r! - \left| \left(H_i^l\right)^{(r)} (\hat{\alpha}^l) \right| /r!}{\left| \left(H_i^l\right)^{(r)} (\hat{\alpha}^l) \right| /r! + \beta}.$$

Similarly to what we have already noticed, the quantity $\left(H_i^l\right)^{(r)} (t),\ t \in [x_i, \max\{x_{i+1}, \bar{x}_{i+1}\}]$, can be expressed by values of a continuous function defined by partial derivatives of f, evaluated at $(t, z_i(t))$, where the argument $(t, z_i(t))$ belongs to the compact set $[a, b] \times D$. By the uniform continuity of this function, we have that

$$\max_{0 \le i \le m-1} \max_{1 \le l \le d} \sup_{\bar{\alpha}, \hat{\alpha}^l \in [x_i, \max\{x_{i+1}, \bar{x}_{i+1}\}]} |\kappa_i^l| \to 0, \quad \text{as } m \to \infty.$$

Hence $|\kappa_i^l| \le \varphi$ for $m \ge \bar{m}_0$, which leads to (25). ∎

Lemma 2 *Let $f \in F_r$. There exist $\bar{C}$, $\tilde{C}$, m_0 such that for any $m \ge m_0$, for any $\{x_i\}_{i=0}^m$ satisfying (4) it holds*

$$\left\| H_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r] - \tilde{H}_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r] \right\| \le \bar{C} \frac{1}{\bar{h}_i^r} \sup_{t \in [x_i, \bar{x}_{i+1}]} \|z_i(t) - \bar{l}_{i,r+1}(t)\| \le \tilde{C} \bar{h}_i, \tag{27}$$

for $i = 0, 1, \ldots, m-1$.
(Here $\bar{t}_k$ are, as above, the equidistant nodes in $[x_i, \bar{x}_{i+1}]$; the index i is omitted.)

Proof The proof follows from the fact that

$$H_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r] - \tilde{H}_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r] = \sum_{k=0}^{r} \frac{f(\bar{t}_k, z_i(\bar{t}_k)) - f(\bar{t}_k, \bar{l}_{i,r+1}(\bar{t}_k))}{\prod\limits_{p=0, p \ne k}^{r} (\bar{t}_k - \bar{t}_p)},$$

and from (18) applied to the approximation $\bar{l}_{i,r+1}$ in the interval $[x_i, \bar{x}_{i+1}]$ with h_i replaced by $\bar{h}_i$. ∎

From (23), Lemmas 1 and 2 we get the following computable (asymptotic) upper bound on the local error of the method ϕ^*.

Theorem 2 *Let $f \in F_r$, $\beta > 0$ and $\varphi \in (0,1)$. There exists m_0 such that for all $m \geq m_0$, for any $\{x_i\}_{i=0}^m$ satisfying (4), and any $\bar{x}_{i+1}$ satisfying (24) it holds*

$$e_{i,r+1} \leq G_i\, h_i^{r+1}, \quad i = 0, 1, \ldots, m-1, \tag{28}$$

where

$$G_i = G_i(f) = (8/3)\left(\|\tilde{H}_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r]\| + \beta\right)(1+\varphi). \tag{29}$$

Proof We successively use (23), (25) and (27). We first get

$$e_{i,r+1} \leq 2\left(\|H_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r]\| + \beta\right)(1+\varphi)\, h_i^{r+1},$$

and next

$$e_{i,r+1} \leq 2\left(\|\tilde{H}_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r]\| + \tilde{C}\bar{h}_i + \beta\right)(1+\varphi)\, h_i^{r+1}.$$

For sufficiently large m such that $\tilde{C}\bar{h}_i \leq \beta/3$ we get (28). ■

For given $x_i < b$ and y_i, after selecting a point $\bar{x}_{i+1}$, we are able to construct an auxillary approximation $\bar{l}_{i,r+1}$ and compute $\tilde{H}_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r]$. Hence, for given β and φ, the coefficient G_i can be effectively computed. For further purposes, note that

$$G_i \geq (8/3)\beta. \tag{30}$$

On the other hand, due to (27), we have in terms of H_i that

$$G_i \leq (8/3)\left(\|H_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r]\| + 1 + \beta\right)(1+\varphi),$$

for sufficiently large m. Due to (26) and the observation made in the proof of Lemma 1 regarding the derivatives of H_i, we have the bound

$$G_i \leq N(f), \tag{31}$$

where $N(f)$ is independent of i and m.

5 Algorithm with guaranteed local error

Let $\varepsilon \in (0,1)$. Our aim is to select mesh points $\{x_i^*\}$ in the method ϕ^* in order to guarantee that in all steps the local error is at most ε,

$$e_{i,r+1} \leq \varepsilon, \qquad i = 0, 1, \ldots, m-1. \tag{32}$$

Given x_i^* and y_i^*, we set

$$\bar{x}_{i+1} = x_i^* + \min\{h(\varepsilon), b - x_i^*\}, \tag{33}$$

where $h : (0,1) \to \mathbf{R}_+$ is any function such that $h(\varepsilon) = O(\varepsilon^{1/(r+1)})$, as $\varepsilon \to 0^+$, with a constant in the $'O'$-notation possibly dependent on f (but not on i or the number of subintervals). We discuss the form of $h(\varepsilon)$ below. We then compute the auxillary approximation $\bar{l}_{i,r+1}$ defined in Section 4 before Lemma 1, and the coefficient G_i from (29), with $[x_i, \bar{x}_{i+1}]$ replaced by $[x_i^*, \bar{x}_{i+1}]$. The mesh point $x_{i+1}^* \le b$ is now selected such that

$$G_i h_i^{r+1} = \varepsilon \quad (h_i = x_{i+1}^* - x_i^*), \tag{34}$$

that is, we put

$$x_{i+1}^* = x_i^* + \min\left\{\left(\frac{\varepsilon}{G_i}\right)^{1/(r+1)}, b - x_i^*\right\}. \tag{35}$$

With this mesh point x_{i+1}^*, the approximation $l_{i,r+1}$ defined in the algorithm ϕ^* satisfies, due to Theorem 2, the condition (32).

Remark 1 We comment on the choice of $h(\varepsilon)$. We can choose $\bar{x}_{i+1}$ 'large' by taking $h(\varepsilon) = \varepsilon^{1/(r+1)}$, so that

$$\bar{x}_{i+1} = x_i^* + \min\left\{\varepsilon^{1/(r+1)}, b - x_i^*\right\}. \tag{36}$$

The condition (24) holds for $\bar{x}_{i+1}$ and x_{i+1}^*, since $x_{i+1}^* - x_i^* = \Theta(\bar{x}_{i+1} - x_i^*)$, which follows from (30) and (31). If we take $h(\varepsilon) = O(\varepsilon)$, then the condition (24) holds for $\bar{x}_{i+1}$ and x_{i+1}^*, since $\bar{x}_{i+1} \le x_{i+1}^*$ for sufficiently small ε. In this case $\bar{x}_{i+1}$ can be arbitrarily close to x_i^*. Our results hold for such $h(\varepsilon)$. It seems that the faster $h(\varepsilon)$ goes to zero with $\varepsilon \to 0$, the earlier the asymptotics shows up.

We have arrived at the following algorithm.

Algorithm ADAPT-MESH

1. Choose $\varepsilon \in (0,1)$, $\beta > 0$ and $\varphi \in (0,1)$. Set $x_0^* = a$ and $y_0^* = \eta$.
2. Given x_i^* and y_i^*, compute $\bar{x}_{i+1}$ from (33). Compute the equidistant points $\bar{t}_0 = x_i^*, \bar{t}_1, \ldots, \bar{t}_r = \bar{x}_{i+1}$ from $[x_i^*, \bar{x}_{i+1}]$.
3. Compute the approximation $\bar{l}_{i,r+1}$ in $[x_i^*, \bar{x}_{i+1}]$ from (6) and (7), based on $\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_{r-1}$.
4. Compute (component by component) the divided difference $\tilde{H}_i[\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_r]$, where $\tilde{H}_i(t) = f(t, \bar{l}_{i,r+1}(t))$.
5. Compute G_i from (29).
6. Compute x_{i+1}^* from (35), and the equidistant points from $[x_i^*, x_{i+1}^*]$ with $t_0 = x_i^*$ for

$r = 1$ and $t_0 = x_i^*, t_1, \dots, t_{r-1} = x_{i+1}^*$ for $r \geq 2$.
7. Compute the approximation $l_{i,r+1}$ in $[x_i^*, x_{i+1}^*]$ from (6) and (7), based on $t_0, t_1, \dots, t_{r-1}$.
8. Set $y_{i+1}^* = l_{i,r+1}(x_{i+1}^*)$. If $x_{i+1}^* < b$, then go to **2** with $i := i + 1$.
STOP

Steps **2**–**5** can be viewed as a 'prediction' stage, while the final approximation is computed in steps **6** and **7**.
We denote by $m^* = m^*(\varepsilon)$ the number of subintervals defined by the algorithm ADAPT-MESH. The local error of the approximation $l_{i,r+1}$ in the interval $[x_i^*, x_{i+1}^*]$ is guaranteed to be at most ε,

$$\sup_{t \in [x_i^*, x_{i+1}^*]} \|z_i(t) - l_{i,r+1}(t)\| \leq \varepsilon, \quad i = 0, 1, \dots, m^* - 1,$$

for sufficiently small ε, see (32). The cost of each step of the method ϕ^*, when applied on a given mesh, is $\kappa^*(r) = r^2 + \Theta(r)$ evaluations of f which are needed to produce $l_{i,r+1}$. The algorithm ADAPT-MESH is additionally equipped with the mesh selection procedure which makes the cost twice as large. The cost of ADAPT-MESH equals $2r^2 + \Theta(r)$ function evaluations per step, which is roughly $2m^*\kappa^*(r)$ in total. In Section 7 we compare it with the cost of other methods and mesh selection procedures.

By the definition of $\{x_i^*\}$, we have that $x_{i+1}^* - x_i^* = (\varepsilon/G_i)^{1/(r+1)}$, $i = 0, 1, \dots, m^* - 2$, and $x_{m^*-1}^* + (\varepsilon/G_{m^*-1})^{1/(r+1)} \geq b$. This yields that m^* is the minimal number $m \in \mathbf{N}$ such that

$$mS(m) \geq (b - a) \left(\frac{1}{\varepsilon}\right)^{1/(r+1)}, \tag{37}$$

where

$$S(m) = \frac{1}{m} \sum_{i=0}^{m-1} \left(\frac{1}{G_i}\right)^{1/(r+1)}. \tag{38}$$

Taking into account the bounds on G_i, we have that

$$\left(\frac{8}{3}\beta\right)^{1/(r+1)} (b - a) \left(\frac{1}{\varepsilon}\right)^{1/(r+1)} \leq m^* < N(f)^{1/(r+1)} (b - a) \left(\frac{1}{\varepsilon}\right)^{1/(r+1)} + 1. \tag{39}$$

We see that the mesh selection procedure in the algorithm ADAPT-MESH does not reduce the speed of growth of the cost as $\varepsilon \to 0$, with respect to the equidistant mesh. As in the latter case (see Section 7), we have that

$$m^*(\varepsilon) = \Theta\left(\left(\frac{1}{\varepsilon}\right)^{1/(r+1)}\right). \tag{40}$$

A potential gain of adaption lies in reducing the coefficient.
Note that the condition (4) is satisfied for $\{x_i^*\}$. Indeed, we have using (39) that

$$\max_{0\le i\le m^*-1}(x_{i+1}^*-x_i^*)\le\left(\frac{3\varepsilon}{8\beta}\right)^{1/(r+1)}\le 2\cdot 3^{1/(r+1)}\left(\frac{N(f)}{8\beta}\right)^{1/(r+1)}\frac{b-a}{m^*},\tag{41}$$

for sufficiently small ε. Hence, (4) holds with any

$$K\ge 2\cdot 3^{1/(r+1)}\left(\frac{N(f)}{8\beta}\right)^{1/(r+1)}\quad\text{and}\quad \ell(m)\ge\frac{b-a}{m}.$$

6 Mesh selection for a general class of methods

The mesh selection procedure described above can be applied to a class of methods ϕ for solving (1), not only for ϕ^*. We assume that for any discretization $\{x_i\}_{i=0}^m$ satisfying (4), a method ϕ successively computes in each interval $[x_i,x_{i+1}]$ an approximation l_i to z_i, with $l_i(x_i)=y_i$ and $y_{i+1}=l_i(x_{i+1})$, starting from $x_0=a$, $y_0=\eta$. Global approximation l computed by ϕ in $[a,b]$ is composed of the approximations l_i in $[x_i,x_{i+1}]$, $\phi(f)(t)=l(t)=l_i(t)$, $t\in[x_i,x_{i+1}]$, $i=0,1,\ldots,m-1$. We assume that the computation of l_i requires a certain number of evaluations of some functionals on f (information about f). The total number of evaluations in a single interval $[x_i,x_{i+1}]$ is $\kappa_\phi(r)$, where $\kappa_\phi(r)$ is independent of i and m. For instance, for the method ϕ^* the functionals are defined by evaluations of f, and $\kappa_{\phi^*}(r)=2r^2+\Theta(r)$. We assume about ϕ that:

A. There are $\bar\beta,\beta>0$ such that for any $f\in F_r$ there is m_0 such that for all $m\ge m_0$, for any $\{x_i\}_{i=0}^m$ satisfying (4)

$$\sup_{t\in[x_i,x_{i+1}]}\|z_i(t)-l_i(t)\|\le\bar\beta\left(\frac{1}{r!}\sup_{\alpha\in[x_i,x_{i+1}]}\|z_i^{(r+1)}(\alpha)\|+\beta\right)h_i^{r+1},\quad i=0,1,\ldots,m-1.\tag{42}$$

Assumption **A** has been verified for $\phi=\phi^*$ in (23). Of course, the method ϕ^* is not the only example of ϕ. It can also be defined in many different ways, e.g., by Taylor's approximation.

Remark 2 It is easy to see that Theorem 1 (with slightly different constant M), Lemma 1, Lemma 2 and Theorem 2 hold for ϕ, with β given in assumption **A**, with $l_{i,r+1}$ replaced by l_i and $\bar l_{i,r+1}$ replaced by $\bar l_i$. The coefficient G_i is now given by

$$G_i=(4/3)\bar\beta\left(\|\tilde H_i[\bar t_0,\bar t_1,\ldots,\bar t_r]\|+\beta\right)(1+\varphi).\tag{43}$$

We have that

$$(4/3)\bar{\beta}\beta \le G_i \le N(f),$$

where $N(f)$ depends on $\bar{\beta}$, and it is independent of i and m.

We now discuss yet another local error bound for ϕ. We show that the local solution z_i in the bound (42) can be replaced, at cost of changing a constant, by the global solution z.

Lemma 3 *For any $f \in F_r$, $\beta > 0$ and $\varphi \in (0,1)$ there is m_0 such that for all $m \ge m_0$, for any $\{x_i\}_{i=0}^m$ satisfying (4) it holds*

$$\frac{1}{r!} \sup_{\alpha \in [x_i, x_{i+1}]} \|z_i^{(r+1)}(\alpha)\| + \beta = \left(\frac{1}{r!} \sup_{\alpha \in [x_i, x_{i+1}]} \|z^{(r+1)}(\alpha)\| + \beta \right)(1+\kappa_i), \tag{44}$$

$i = 0, 1, \ldots, m-1$, for some κ_i, where $|\kappa_i| \le \varphi$.

Proof Let the two sup above be achieved in points $t_1, t_2 \in [x_i, x_{i+1}]$, respectively. We have

$$\frac{1}{r!}\|z_i^{(r+1)}(t_1)\| + \beta = \left(\frac{1}{r!}\|z^{(r+1)}(t_2)\| + \beta \right)(1+\kappa_i),$$

where $\kappa_i = (1/r!)\left(\|z_i^{(r+1)}(t_1)\| - \|z^{(r+1)}(t_2)\|\right) / \left((1/r!)\|z^{(r+1)}(t_2)\| + \beta\right)$. Hence,

$$|\kappa_i| \le \frac{1}{\beta r!}\|z_i^{(r+1)}(t_1) - z^{(r+1)}(t_2)\| \le \frac{1}{\beta r!}\left(\|z_i^{(r+1)}(t_1) - z^{(r+1)}(t_1)\| + \|z^{(r+1)}(t_2) - z^{(r+1)}(t_1)\|\right).$$

The last term is bounded by $\sup_{|t_1 - t_2| \le h_i} \|z^{(r+1)}(t_1) - z^{(r+1)}(t_2)\|$, so that, due to the uniform continuity of $z^{(r+1)}$, it tends to 0 (uniformly with respect to i) as $m \to \infty$. Note that $z_i^{(r+1)}(t_1)$ and $z^{(r+1)}(t_1)$ can be expressed by partial derivatives of f of order $0, 1, \ldots, r$ evaluated in $(t_1, z_i(t_1))$ and $(t_1, z(t_1))$, respectively. Due to Theorem 1 for ϕ, we have

$$\|z_i(t_1) - z(t_1)\| \le \exp(Lh_i)\|y_i - z(x_i)\| = O\left(\max_{0 \le j \le m-1} h_j^r \right).$$

Hence, $z(t_1), z_i(t_1) \in D$, see (3), for sufficiently large m. This and the uniform continuity of the partial derivatives of f in $[a,b] \times D$ yield that the first term also tends to 0 as $m \to \infty$, uniformly with respect to t_1 and i. Hence, $|\kappa_i|$ tends to 0 as $m \to \infty$, uniformly with respect to t_1, t_2 and i. This proves the lemma. ■

We now list upper bounds on the local error of ϕ that appeared so far. The basic one is given in assumption **A**

$$\sup_{t \in [x_i, x_{i+1}]} \|z_i(t) - l_i(t)\| \le c_i\, h_i^{r+1}, \tag{45}$$

where

$$c_i = \bar{\beta}\left(\frac{1}{r!}\sup_{\alpha\in[x_i,x_{i+1}]}\|z_i^{(r+1)}(\alpha)\| + \beta\right). \tag{46}$$

The bound (45) involves the local solution z_i; it usually appears in the error analysis of a method ϕ. The second one follows from Lemma 3 and has the form (45) with c_i replaced by

$$\bar{c}_i = \bar{\beta}\left(\frac{1}{r!}\sup_{\alpha\in[x_i,x_{i+1}]}\|z^{(r+1)}(\alpha)\| + \beta\right)(1+\varphi). \tag{47}$$

We observe that the bounds $\bar{c}_i\, h_i^{r+1}$, $i = 0, 1, \ldots, m-1$, only depend on the local behavior of f and on the mesh $\{x_i\}$. They hold for any method ϕ satisfying **A**, and are useful for theoretical reasons. Note that the function $p(x_i, x_{i+1}) = \bar{c}_i\, h_i^{r+1}$ is an increasing function with respect to x_{i+1} (for fixed x_i), and a decreasing function with respect to x_i (for fixed x_{i+1}).

The third bound is constructive and it will be used in the algorithm ADAPT-MESH-GEN below. It is given by (45) with c_i replaced by G_i from (43). Note that the coefficients (47) and (43) are not overestimated compared to (46); they are equivalent up to a coefficient only dependent on φ, for sufficiently large m. It follows from Lemmas 1, 2 and 3 that for any $m \geq m_0$, any $\{x_i\}_{i=0}^m$ satisfying (4) and $i = 0, 1, \ldots, m-1$ it holds

$$\frac{1-\varphi}{1+\varphi}\,\bar{c}_i \leq c_i \leq \bar{c}_i \quad \text{and} \quad \frac{1-\varphi}{2(1+\varphi)}\,G_i \leq c_i \leq G_i. \tag{48}$$

Hence, for a given φ, the three error bounds

$$c_i\, h_i^{r+1}, \quad \bar{c}_i\, h_i^{r+1}, \quad \text{and} \quad G_i\, h_i^{r+1},$$

are equivalent up to absolute constants (for a fixed φ). In particular, they all reflect a local behavior of f.

As in the case of the method ϕ^*, we are free to choose the mesh points for ϕ. The following algorithm, very much similar to ADAPT-MESH, describes the mesh selection for ϕ that allows us to keep the local error at level ε.

Algorithm ADAPT-MESH-GEN

1. Choose $\varepsilon \in (0,1)$, and $\varphi \in (0,1)$. Set $x_0 = a$ and $y_0 = \eta$.

2. Given x_i and y_i, compute

$$\bar{x}_{i+1} = x_i + \min\{h(\varepsilon), b - x_i\}. \tag{49}$$

3. Compute an approximation $\bar{l}_i$ in $[x_i, \bar{x}_{i+1}]$ using ϕ.
4. For $\tilde{H}_i(t) = f(t, \bar{l}_i(t))$, compute (component by component) the divided difference $\tilde{H}_i[\bar{t}_0, \bar{t}_1, \dots, \bar{t}_r]$, where $\bar{t}_0 = x_i, \bar{t}_1, \dots, \bar{t}_r = \bar{x}_{i+1}$ are the equidistant points from $[x_i, \bar{x}_{i+1}]$.
5. Compute $G_i = (4/3)\bar{\beta}\left(\|\tilde{H}_i[\bar{t}_0, \bar{t}_1, \dots, \bar{t}_r]\| + \beta\right)(1+\varphi)$.
6. Compute

$$x_{i+1} = x_i + \min\left\{\left(\frac{\varepsilon}{G_i}\right)^{1/(r+1)}, b - x_i\right\}. \tag{50}$$

7. Compute an approximation l_i in $[x_i, x_{i+1}]$ using ϕ.
8. Set $y_{i+1} = l_i(x_{i+1})$. If $x_{i+1} < b$, then go to **2** with $i := i+1$.
STOP

Since $c_i\, h_i^{r+1} \le G_i\, h_i^{r+1} \le \varepsilon$, we see that the local error of the approximation l_i computed in the step **7** is guaranteed to be at most ε,

$$\sup_{t\in[x_i,x_{i+1}]} \|z_i(t) - l_i(t)\| \le \varepsilon, \quad i = 0, 1, \dots, m-1, \tag{51}$$

for sufficiently small ε. The cost of each step of ADAPT-MESH-GEN, measured by the number of evaluations of functionals on f needed to produce l_i, is doubled with respect to the cost of ϕ applied on a given mesh.

7 Adaptive mesh – the cost analysis

Consider an arbitrary method ϕ satisfying **A**, based on a mesh $x_0 = a < x_1 < \ldots < x_m = b$ for which (4) holds. We measure the cost of ϕ, $\mathrm{cost}(\phi, m)$, by the total number of evaluations of functionals on f needed for computing l in all subintervals $[x_i, x_{i+1}]$, i.e.,

$$\mathrm{cost}(\phi, m) = \kappa_\phi(r)\, m. \tag{52}$$

Our goal is to keep the local error at a prescribed level ε, see (51). Assuming that the goal is achieved by some ϕ with some mesh $x_0, x_1, \dots, x_m$, we wish to compare $\mathrm{cost}(\phi, m)$ with the cost of the algorithm ADAPT-MESH.

To compare the costs of algorithms, we use the local error bound $c_i\, h_i^{r+1}$ from assumption **A**. We wish to assure that

$$c_i\, h_i^{r+1} \le \varepsilon, \qquad \text{for all } i, \tag{53}$$

which implies (51). We define the reference quantity $\hat{m}(\varepsilon)$ as follows. Let

$$\hat{k}(m) = \inf\left\{ \max_{0\le i\le m-1} c_i\, h_i^{r+1} : \quad x_0 = a \le x_1 \le \ldots \le x_{m-1} \le x_m = b \ \text{ satisfies (4)} \right\}. \tag{54}$$

Then we define

$$\hat{m}(\varepsilon) = \min\left\{ m \in \mathbf{N} : \ \ \hat{k}(m) \le \varepsilon \right\}. \tag{55}$$

Thus, $\hat{m}(\varepsilon)$ is the minimal number of subintervals m for which there exists a mesh with $m+1$ points such that $c_i\, h_i^{r+1} \le \varepsilon$, $i = 0, 1, \ldots, m-1$.
Define similarly to (54) and (55) the following, technically useful, quantities

$$\bar{k}(m) = \inf\left\{ \max_{0\le i\le m-1} \bar{c}_i\, h_i^{r+1} : \quad x_0 = a \le x_1 \le \ldots \le x_{m-1} \le x_m = b \ \text{ satisfies (4)} \right\} \tag{56}$$

and

$$\bar{m}(\varepsilon) = \min\left\{ m \in \mathbf{N} : \ \ \bar{k}(m) \le \varepsilon \right\}. \tag{57}$$

Since $\bar{c}_i\, h_i^{r+1}$ is an increasing function of x_{i+1} (for fixed x_i) and a decreasing function of x_i (for fixed x_{i+1}), for sufficiently small $\varepsilon > 0$ the quantity $\bar{m}(\varepsilon)$ can be computed as follows. We start with $\hat{x}_0 = a$, and for a given $\hat{x}_i$, we compute $\hat{x}_{i+1}$ as the unique solution of $\bar{c}_i\,(x_{i+1} - \hat{x}_i)^{r+1} = \varepsilon$. Then $\bar{m}(\varepsilon)$ is the minimal i such that $\hat{x}_i \ge b$.

Note further that for any $m \ge m_0$ and $\{x_i\}_{i=0}^m$ satisfying (4), it follows from (48) that

$$\max_{0\le i\le m-1} c_i\, h_i^{r+1} \le \max_{0\le i\le m-1} \bar{c}_i\, h_i^{r+1} \le \frac{1+\varphi}{1-\varphi} \max_{0\le i\le m-1} c_i\, h_i^{r+1},$$

which yields for $m \ge m_0$ that

$$\hat{k}(m) \le \bar{k}(m) \le \frac{1+\varphi}{1-\varphi}\hat{k}(m). \tag{58}$$

Hence, for any $\varepsilon \in (0, \varepsilon_0]$ (and fixed ϕ, $\bar{\beta}$)

$$\hat{m}\,(\varepsilon) \le \bar{m}(\varepsilon) \le \hat{m}\left(\frac{1-\varphi}{1+\varphi}\,\varepsilon\right). \tag{59}$$

We now compare the cost of ADAPT-MESH with the cost of other algorithms ϕ equipped with any mesh selection procedure. The number of subintervals computed by ADAPT-MESH is $m^*(\varepsilon)$ and the cost of producing $l_{i,r+1}$ in each subinterval is $\kappa^*(r)$. Since the cost is doubled due to the mesh selection, it holds

$$\text{cost}(\phi^*, m^*(\varepsilon)) = 2\kappa^*(r)m^*(\varepsilon). \tag{60}$$

The quantities $\hat{k}(m)$ and $\hat{m}(\varepsilon)$ depend on ϕ (and $\bar{\beta}$). In the following result, we shall use for clarity the notation $\hat{k}_\phi(m)$ and $\hat{m}_\phi(\varepsilon)$. We compare $m^*(\varepsilon)$ from ADAPT-MESH (where $\bar{\beta} = 2$) with the minimal number of intervals for any other method ϕ with $\bar{\beta} = 2$, equipped with any mesh selection strategy. We have

Theorem 3 *Let $f \in F_r$, $\varphi \in (0,1)$ and ϕ be any method satisfying* **A** *with $\bar{\beta} = 2$. Then there exists $\varepsilon_0 \in (0,1)$ such that for any $0 < \varepsilon \le \varepsilon_0$ it holds*

$$\hat{m}_\phi(p_1\varepsilon) \le \hat{m}_{\phi^*}(\varepsilon) \le m^*(\varepsilon) \le \hat{m}_\phi\left(p\varepsilon\right), \tag{61}$$

with $p_1 = (1+\varphi)/(1-\varphi)$ and $p = (1-\varphi)^2/(2(1+\varphi)^2)$.
Hence,

$$\frac{2\kappa^*(r)}{\kappa_\phi(r)} \operatorname{cost}\left(\phi, \hat{m}_\phi\left(p_1\varepsilon\right)\right) \le \operatorname{cost}(\phi^*, m^*(\varepsilon)) \le \frac{2\kappa^*(r)}{\kappa_\phi(r)} \operatorname{cost}\left(\phi, \hat{m}_\phi\left(p\varepsilon\right)\right). \tag{62}$$

Proof The algorithm ADAPT-MESH defines m^*+1 points x_i^* such that for $h_i = x_{i+1}^* - x_i^*$ and G_i for ϕ^* we have

$$G_i\, h_i^{r+1} = \varepsilon, \quad i = 0, 1, \ldots, m^*(\varepsilon) - 2, \qquad \text{and} \qquad G_{m^*-1}\, h_{m^*-1}^{r+1} \le \varepsilon. \tag{63}$$

We show the lower bound in (61). For $\{x_i^*\}_{i=0}^{m^*}$ and ε sufficiently small, we have that $c_i\, h_i^{r+1} \le G_i\, h_i^{r+1} \le \varepsilon$ for $i = 0, 1, \ldots, m^* - 1$. This yields that $\hat{k}_{\phi^*}(m^*) \le \varepsilon$, which implies that $\hat{m}_{\phi^*}(\varepsilon) \le m^*(\varepsilon)$. The further lower bound follows from (59).
We now show the upper bound. By (48), for any $m \ge m_0$, any ϕ, any $\{x_i\}_{i=0}^m$ satisfying (4) and any i, we have for $h_i = x_{i+1} - x_i$ (and G_i for ϕ) that

$$G_i\, h_i^{r+1} \le \frac{2(1+\varphi)}{1-\varphi}\, \bar{c}_i\, h_i^{r+1}.$$

For $\phi = \phi^*$, $m = m^*$ and the mesh $\{x_i^*\}$ given by ADAPT-MESH, due to (63), we have for $h_i = x_{i+1}^* - x_i^*$ and ε sufficiently small that

$$\varepsilon \le \frac{2(1+\varphi)}{1-\varphi}\, \bar{c}_i\, h_i^{r+1}, \quad i = 0, 1, \ldots, m^* - 2,$$

that is,

$$\frac{1-\varphi}{2(1+\varphi)}\, \varepsilon \le \bar{c}_i\, h_i^{r+1}, \quad i = 0, 1, \ldots, m^* - 2. \tag{64}$$

The observation made after (57) yields that the points $\hat{x}_i$ computed for the accuracy $(1-\varphi)/(2(1+\varphi))\varepsilon$ satisfy in the light of (64) the inequalities $\hat{x}_i \le x_i^*$ for $i = 0, 1, \ldots, m^* - 1$. This implies that

$$m^*(\varepsilon) \le \bar{m}\left(\frac{1-\varphi}{2(1+\varphi)}\,\varepsilon\right). \tag{65}$$

Finally, (59) and (65) give us the desired inequality (61)

$$m^*(\varepsilon) \le \hat{m}_\phi\left(\frac{(1-\varphi)^2}{2(1+\varphi)^2}\,\varepsilon\right). \tag{66}$$

The inequalities for the cost follow immediately. ■

Theorem 3 says that the cost of ADAPT-MESH can only exceed by the constant $2\kappa^*(r)/\kappa_\phi(r)$ the cost of any algorithm ϕ with $\bar{\beta} = 2$, with any mesh selection strategy such that the local error is at most $(1-\varphi)^2/(2(1+\varphi)^2)\,\varepsilon$. This accuracy is more demanding than ε. For instance, if we take $\varphi = 1/2$, then ε in the accuracy demand is replaced by $\varepsilon/18$.

Observe that adaptive mesh cannot reduce the speed of growth of the cost as $\varepsilon \to 0$. It follows from (61) and (40) that for the best choice of points we have, similarly as for the equidistant mesh, that

$$\hat{m}(\varepsilon) = \Theta\left(\left(\frac{1}{\varepsilon}\right)^{1/(r+1)}\right).$$

The asymptotics is thus the same. Possible advantage of the adaptive mesh selection is hidden in the size of the quantity $\hat{m}(\varepsilon)$, see the definition (55). To illustrate this, we now discuss possible advantage of the algorithm ADAPT-MESH with respect to another algorithm ϕ with $\bar{\beta} = 2$, based on the uniform mesh. Note that for any ϕ, any m and the uniform mesh we have

$$\max_{0\le i\le m-1} \bar{c}_i\, h_i^{r+1} = \bar{N}(f)\left(\frac{b-a}{m}\right)^{r+1},$$

where $\bar{N}(f) = \bar{\beta}\left((1/r!)\sup_{t\in[a,b]}\|z^{(r+1)}(t)\| + \beta\right)(1+\varphi)$. Hence,

$$\bar{m}^{\mathrm{equid}}(\varepsilon) = \min\left\{m:\ \ \bar{N}(f)\left(\frac{b-a}{m}\right)^{r+1} \le \varepsilon\right\} = \left\lceil (b-a)\left(\frac{\bar{N}(f)}{\varepsilon}\right)^{1/(r+1)}\right\rceil. \tag{67}$$

It follows from (48) that we also have

$$\max_{0\le i\le m-1} c_i\, h_i^{r+1} = \Theta\left(\bar{N}(f)\left(\frac{b-a}{m}\right)^{r+1}\right), \qquad \text{as } m\to\infty,$$

and

$$\hat{m}^{\text{equid}}(\varepsilon) = \Theta\left(\left\lceil (b-a)\left(\frac{\bar{N}(f)}{\varepsilon}\right)^{1/(r+1)}\right\rceil\right), \quad \text{as } \varepsilon \to 0,$$

where constants in the Θ-notation only depend on φ.
We want to compare $m^*(\varepsilon)$ (which decides about the cost of ADAPT-MESH) with $\hat{m}^{\text{equid}}(\varepsilon)$ (which decides about the cost of ϕ with the equidistant mesh). We have by (65) the following sequence of inequalities

$$\begin{aligned} m^*(\varepsilon) &\leq \bar{m}\left(\frac{1-\varphi}{2(1+\varphi)}\,\varepsilon\right) \leq \bar{m}^{\text{equid}}\left(\frac{1-\varphi}{2(1+\varphi)}\,\varepsilon\right) \\ &= \Theta\left(\left\lceil (b-a)\left(\frac{\bar{N}(f)}{\varepsilon}\right)^{1/(r+1)}\right\rceil\right) = \Theta\left(\hat{m}^{\text{equid}}(\varepsilon)\right), \end{aligned} \tag{68}$$

where again constants in the Θ-notation only depend on φ.
The second inequality in (68) allows us to understand when the adaption pays off. Just below the definition (57), we gave a comment on how to compute $\bar{m}(\varepsilon)$. The comment yields that $\bar{m}(\varepsilon)$ is the minimal number $m \in \mathbf{N}$ such that

$$m\bar{S}(m) \geq (b-a)\left(\frac{1}{\varepsilon}\right)^{1/(r+1)}, \tag{69}$$

where

$$\bar{S}(m) = \frac{1}{m}\sum_{i=0}^{m-1}\left(\frac{1}{\bar{c}_i}\right)^{1/(r+1)}. \tag{70}$$

To see this, consult similar reasoning leading to (37) and (38). In the case of $\bar{m}^{\text{equid}}(\varepsilon)$ an analogous condition to (69) is given in the first equality in (67):

$$m\frac{1}{\bar{N}(f)^{1/(r+1)}} \geq (b-a)\left(\frac{1}{\varepsilon}\right)^{1/(r+1)}. \tag{71}$$

Comparing (69) and (71), we see that the second term in (68) is much less than the third term if $\bar{S}(m)^{-1}$ is much smaller than $\bar{N}(f)^{1/(r+1)}$. Since

$$\bar{N}(f) = \bar{\beta}\left(\frac{1}{r!}\sup_{t\in[a,b]}\|z^{(r+1)}(t)\| + \beta\right)(1+\varphi),$$

and

$$\bar{c}_i = \bar{\beta}\left(\frac{1}{r!}\sup_{t\in[x_i,x_{i+1}]}\|z^{(r+1)}(t)\| + \beta\right)(1+\varphi),$$

we can identify cases when the gain of adaption is significant. Adaption pays off for functions for which the size of $\|z^{(r+1)}(t)\|$ changes significantly in parts of the interval $[a,b]$. Of course, the second inequality in (68) can also turn into equality. For such functions f there is no gain of adaption. Translating the above discussion to similar properties of the cost of the algorithm is straightforward.

8 Numerical example

We illustrate the performance of the mesh selection mechanism in ADAPT-MESH by an example (other test examples in C^{++} are in progress), see [11]. We consider a scalar test problem from [4] with a parameter $\delta > 0$

$$z'(t) = \frac{3}{4}(z(t) - 1)^{-3/2}, \;\; t \in [0,1], \quad z(0) = 1 + \delta, \tag{72}$$

with the global solution given by $z(t) = \left(\frac{15}{8}t + \delta^{5/2}\right)^{2/5} + 1$. The solution with the initial condition $z(x) = y \quad (x \geq 0,\, y > 1)$ is given by

$$z_{x,y}(t) = \left(\frac{15}{8}(t - x) + (y - 1)^{5/2}\right)^{2/5} + 1.$$

The right-hand side $f(t, y) = \frac{3}{4}(y - 1)^{-3/2}$ is a C^∞ function for $y > 1$.
The problem (72) is a typical test problem whose computational difficulty can be controlled by δ; it grows as δ tends to zero. We use the algorithm ADAPT-MESH with $r = 1$, which corresponds to the Euler method equipped with the mesh selection algorithm, and with $r = 2$. For the global solution z, we have that $|z^{(r+1)}(t)|$ for t close to 0 behaves like $1/\delta^4$ for $r = 1$ and $1/\delta^{6.5}$ for $r = 2$. For t away from 0, $|z^{(r+1)}(t)|$ is essentially a constant. That is, for small δ we should observe a significant advantage of adaptive mesh points over the equidistant points.
The computer precision is $\mathit{macheps} = 10^{-15}$. Obviously, since $\mathit{macheps}$ is fixed and computing time is limited, we cannot verify the asymptotic behavior of the algorithm as $\varepsilon \to 0$; we are only able to see results for some number of values of ε.
Let us now briefly discuss a practical choice of $h(\varepsilon)$ in step **2** of the algorithm. In fixed precision computation, the crucial point is accuracy of computing the divided difference in (27) of Lemma 2. Due to round off errors in computing both $f(\bar{t}_k, \bar{l}_{i,r+1}(\bar{t}_k))$ and the divided difference, the bound (27) changes to $\tilde{C}(\bar{h}_i + \mathit{macheps}/\bar{h}_i^r)$, for some $\tilde{C}$ dependent on f. The minimum of the function of $\bar{h}_i$ is achieved for $\bar{h}_i = (r \cdot \mathit{macheps})^{1/(r+1)}$. Thus, in step **2**, neglecting the coefficient dependent on r, we fix $h(\varepsilon)$ independently of ε to be $h(\varepsilon) = 10^{-15/(r+1)}$.

In step **5**, we set

$$G_i = \begin{cases} 2|\tilde{H}_i[x_i^*, \bar{x}_{i+1}]| + 1 & r = 1, \\ 4\left|\tilde{H}_i[x_i^*, (x_i^* + \bar{x}_{i+1})/2, \bar{x}_{i+1}]\right| + 2 & r = 2. \end{cases}$$

(Note that for $r = 1$ we have $l_{i,r+1} = \bar{l}_{i,r+1}$.) The following table shows results computed by ADAPT-MESH for a number of values of δ and ε. We denote

$$\text{MAXERR} = \max_{0 \le i \le m^*-1} |z_i(x_{i+1}^*) - y_{i+1}^*|,$$

where $z_i(x_i^*) = y_i^*$, and m^* is the number of intervals computed in ADAPT-MESH. The value EQUIDIST is the maximal local error of the respective method for $r = 1$ or $r = 2$ applied on the equidistant mesh $x_i = i/m^*$, $i = 0, 1, \ldots, m^*$, with the same number of subintervals equal to m^*. In the successive columns we show the values m^* , MAXERR/ε and EQUIDIST/ε.

δ	ε	m^* $r=1$	MAXERR/ε $r=1$	EQUIDIST/ε $r=1$	m^* $r=2$	MAXERR/ε $r=2$	EQUIDIST/ε $r=2$
0.1	10^{-2}	33	0.22	49.42	24	0.03	26.06
	10^{-4}	315	0.246	225.7	99	0.04	345.62
	10^{-8}	31373	0.25	424.4	2081	0.04	5331.38
	10^{-14}	31371619	0.264	428.7	207780	0.06	7409.35
10^{-2}	10^{-2}	41	0.22	1801.15	33	0.04	1105.64
	10^{-4}	390	0.25	18147.4	136	0.11	25876.9
	10^{-8}	38841	0.25	907049	2821	0.16	$9.15 * 10^6$
	10^{-14}	38839361	0.26	$2.79 * 10^6$	281583	0.175	$4.73 * 10^9$
10^{-3}	10^{-2}	43	0.22	55127.5	32	1.3	37025.9
	10^{-4}	413	0.37	$5.73 * 10^5$	140	18.65	$8.45 * 10^5$
	10^{-8}	41109	0.49	$5.6 * 10^7$	2917	950.194	$4.0 * 10^8$
	10^{-14}	41106703	0.5	$1.48 * 10^{10}$	291133	2262.01	$3.35 * 10^{12}$
10^{-4}	10^{-2}	42	1.005	$1.79 * 10^6$	22	16.14	$1.7 * 10^6$
	10^{-4}	414	8.09	$1.81 * 10^7$	121	336.5	$3.1 * 10^7$
	10^{-8}	41367	77.96	$1.81 * 10^9$	2915	118505	$1.29 * 10^{10}$
	10^{-14}	41365164	88.96	$1.71 * 10^{12}$	291276	$7.88 * 10^7$	$1.28 * 10^{14}$

According to the theory, for sufficiently small ε the values in the 4th column (for $r = 1$) and 7th column (for $r = 2$) should be at most 1. This is the case for $\delta = 10^{-1}, 10^{-2}$ for $r = 1, 2$,

and $\delta = 10^{-3}$ for $r = 1$. For small values of δ, the round off errors do not allow us to observe the asymptotic behavior of the algorithm, since the value of ε is too large. Comparison of columns 4 and 5 for $r = 1$ and 7 and 8 for $r = 2$ shows the gain of the adaptive mesh selection algorithm applied in ADAPT-MESH over the equidistant points. In the test we have computed results for the equidistant mesh with the same number of points. We may wish to compare the behavior of adaption with nonadaption using the same number of evaluations of f. For $r = 1$, the adaptive method uses 2 function evaluations, while the nonadaptive one only one value. Hence, in this case the value in the 5th column should be divided by 4. For $r = 2$, the respective numbers are 10 and 4 evaluations, that is the result in the 8th column should be divided roughly by 16. This does not change the picture – in both cases, for small δ the tests show a very significant advantage of the adaption over nonadaption.
We shortly comment on comparison between the algorithm defined in [4] for scalar autonomous problems and the current algorithm designed for systems of IVPs, for the test problem (72). As it can be expected, the algorithm from [4] allows us to better treat small values of δ. This follows from the fact that, roughly speaking, the step size control in [4] was based on two-sided estimates of local errors. Specific properties of scalar autonomous problems were used in [4]; they cannot be extended to systems of initial value problems. In order to handle systems of IVPs, the present algorithm uses upper local error bounds, see (28) and (29).

9 Conclusions

We have proposed a mesh selection algorithm for systems of IVPs that (asymptotically) guarantees a given level of the local error. The algorithm only requires evaluations of the right-hand side f. Rigorous analysis of the cost has been given, including comparison with the best choice of the mesh points, as well as with the uniform mesh.

References

[1] Choi, S.T., Ding, Y., Hickernell, F.J, Tong, X. (2017), Local adaption for approximation and minimization of univariate functions, *J. Complexity*, 40, 17–33, http://dx.doi.org/10.1016/j.jco.2016.11.005.

[2] Daun, T., (2011), On the randomized solution of initial value problems, *J. Complexity*, 27, 300–311.

[3] Jackiewicz, Z. (2002), Implementation of DIMSIMs for stiff differential systems, *Appl. Numer. Math.*, 42, No. 1-3, 251–267.

[4] Kacewicz, B., (2017), Adaptive mesh point selection for the efficient solution of scalar IVPs, to appear in *Numerical Algorithms*, DOI: 10.1007/s11075-017-0304-2

[5] Kacewicz, B., Przybyłowicz, P., (2015), Complexity of the derivative-free solution of systems of IVPs with unknown singularity hypersurface, *J. Complexity*, 31, 75–97.

[6] Lyness, J.N., (1983), When not to use an automatic quadrature routine?, *SIAM Review*, 25, 63–87.

[7] Mazzia, F., Nagy, A.M, (2015), A new mesh selection strategy with stiffness detection for explicit Runge Kutta methods, *Applied Math. and Comp.*, 255, 125–134.

[8] Piessens, R., De Doncker-Kapenga,E., Überhuber,C. W., (1983), *QUADPACK: a subroutine package for automatic integration*, Springer, ISBN: 3-540-12553-1.

[9] Plaskota, L., (2015), Automatic integration using asymptotically optimal adaptive Simpson quadrature, *Numer. Math.*, 131, 173–198.

[10] Plaskota, L., Wasilkowski, G.W., (2009), The power of adaptive algorithms for functions with singularities, *J. Fixed Point Theory Appl.*, 6, 227–248.

[11] Wilga, B., (2017), Adaptive mesh selection in the numerical solution of IVPs, in Polish, master's thesis supervised by B. Kacewicz, Faculty of Applied Mathematics, AGH University of Science and Technology, Cracow, Poland, in preparation.