involve
a journal of mathematics

An explicit third-order one-step method for autonomous scalar initial value problems of first order based on quadratic Taylor approximation

Thomas Krainer and Chenzhang Zhou

msp

2020 vol. 13, no. 2

INVOLVE 13:2 (2020)
dx.doi.org/10.2140/involve.2020.13.231

An explicit third-order one-step method for autonomous scalar initial value problems of first order based on quadratic Taylor approximation

Thomas Krainer and Chenzhang Zhou

(Communicated by Kenneth S. Berenhaut)

We present an explicit one-step numerical method of third order that is error-free on autonomous scalar Riccati equations such as the logistic equation. The method replaces the differential equation by its quadratic Taylor polynomial in each step and utilizes the exact solution of that equation for the calculation of the next approximation.

1. Introduction

One of the basic ordinary differential equations in quantitative population dynamics is the logistic differential equation

$$\begin{cases} \dot{y} = ry\left(1 - \dfrac{y}{K}\right) - qy, \\ y|_{t=0} = y_0, \end{cases}$$

where $y = y(t)$ is the size of the population at time $t \geq 0$, $r > 0$ is the maximal growth rate for the population, and $K > 0$ is the carrying capacity of the habitat for the population under study. We modified the equation in this example by a harvesting term with harvesting rate $q > 0$ as it appears, for example, in fishery models. We refer to [Thieme 2003] as a general reference for ordinary differential equation models in ecology. While the logistic model, as well as its variations and perturbations, are classical cornerstones of ecological quantitative modeling, it is remarkable that the standard numerical methods for approximating solutions to ordinary differential equations do not solve the logistic equation error-free. Motivated by this observation, we are presenting here an explicit third-order one-step numerical method that is applicable to scalar autonomous initial value problems of the form

$$\begin{cases} \dot{y} = f(y), \\ y|_{t=0} = y_0, \end{cases} \tag{1.1}$$

MSC2010: 65L05.
Keywords: numerical methods for ODEs, initial value problems.

with sufficiently smooth real-valued f that approximates the solution $y = y(t)$ on the compact interval $[0, T]$ by a sequence of values $y_0, y_1, y_2, \ldots$ based on equidistant time-stepping with step size $h > 0$, and whose distinguishing feature is that the method is error-free if f is a polynomial up to degree 2 such as in the logistic equation; i.e., our method solves autonomous Riccati equations exactly. The idea for this method is simple:

(1) Replace $f(y)$ in (1.1) by its quadratic Taylor polynomial $T_{y_0}(y)$ centered at y_0.

(2) Solve $\dot{u} = T_{y_0}(u)$ exactly with initial condition $u(0) = y_0$.

(3) Set $y_1 = u(h)$ and repeat with y_1 in place of y_0, etc.

There are several issues that arise upon implementation of this basic idea. Most importantly, it must be noted that solutions to Riccati equations can blow up in finite time, so integrity checks on the step size $h > 0$ are needed to preclude a potential blow-up of the approximate solution u on the interval $(0, h]$ as otherwise the calculated term y_1 is invalid (and likewise in subsequent steps). To make this more transparent, consider the example

$$\begin{cases} \dot{y} = (y - \lambda)(1 - y)e^{-y^4}, \\ y|_{t=0} = 0, \end{cases}$$

where $\lambda \gg 1$. The maximal solution to this differential equation exists on $(-\infty, \infty)$ because the function

$$f(y) = (y - \lambda)(1 - y)e^{-y^4}, \quad -\infty < y < \infty,$$

is bounded. However, the maximal solution to $\dot{u} = T_0(u) = (u-\lambda)(1-u)$ with initial value $u(0) = 0$ blows up at $t = \ln(\lambda)/(\lambda - 1)$. In view of $\lim_{\lambda\to\infty} \ln(\lambda)/(\lambda - 1) = 0$ we see that blow-up does occur on $(0, h)$ if $\lambda \gg 1$ is large enough.

We provide two options to deal with this problem in the implementation, a priori or at run time. The a priori option calculates a threshold for how small the step size $h > 0$ ought to be chosen at the outset to avoid invalid approximating terms throughout and is based on the differential equation (1.1) and the viewing window $[0, T] \times [y_{\min}, y_{\max}]$ where its solution is supposed to be approximated as inputs, while the run time option checks validity of each approximation value at the time when it is calculated. The issue of blow-up is germane to the method we discuss in this note; it does not occur in standard Runge–Kutta methods or exponential integrators.

A second issue that we needed to address in the implementation concerns evaluation of the formula for the approximate solution u itself. If the roots of the Taylor polynomial T_{y_0} are distinct but close, the exact formula for u would require evaluation of quotients nearly of the form $0/0$ (but with defined limiting value corresponding to the double-root case). We deal with this by introducing a tolerance

parameter $0 < \mathrm{tol}_0 \ll 1$ and replace evaluation of the exact formula for u by appropriate expansions once the critical expressions fall under tolerable thresholds. Problems of similar kind are well known to arise elsewhere in numerical ODEs, for example in exponential integrators where evaluation of $\phi_1(z) = (e^z - 1)/z$ for z near zero occurs; see [Hochbruck and Ostermann 2010; Kassam and Trefethen 2005].

The general idea of utilizing zeroth- and first-order Taylor approximations in the differential equation is well-established both in theoretical and computational ODEs. In computational ODEs, adaptive first-order Taylor approximation (linearization) is the basis for exponential integrators, classically rooted in the Rosenbrock–Euler method (observe that adaptive zeroth-order Taylor approximation in the differential equation yields the Euler method). Exponential integrators [Hochbruck and Ostermann 2010] have been widely used for stiff problems over the past 30 years; they are generally more effective for these problems than standard Runge–Kutta methods because the linearization of the differential equation is solved exactly. Since the theoretical underpinning for exponential integrators is linear theory, they have been developed into a versatile family of methods applicable to single equations and systems alike. The method we present in this note based on adaptive quadratic Taylor approximation of the differential equation is qualitatively more accurate than methods rooted in linearization, but does not exhibit the same degree of versatility and universality, and the applicability is strictly limited to autonomous scalar equations. The reason is that ordinary differential equations with quadratic nonlinearities generally do not allow for closed solution formulas, the autonomous case of a single unknown function being an exception.

We note that our work relates to nonstandard finite difference models and their applications to numerical ODEs as pioneered by Mickens [1994; 2000]; see also [Patidar 2005]. In particular, exact nonstandard finite difference models for the logistic equation and many other ODEs where explicit solution formulas are available are well known [Vigo-Aguiar and Ramos 2011].

The paper is structured as follows: Section 2 covers the theoretical part. We prove, more generally than what has been stated above, that when the function f in (1.1) is adaptively replaced by its r-th order Taylor polynomial and the exact solution to the modified ODE is used to calculate the next approximating value, we obtain a well-defined convergent explicit numerical method of order $r + 1$. More precisely, when the exact solution is supposed to be approximated in the window $[0, T] \times [y_{\min}, y_{\max}]$, we show that there is a threshold $h_0 > 0$ such that the method is defined everywhere in that window for step sizes $0 < h < h_0$ and allows calculation of the next approximating value to the solution. This qualitatively addresses the aforementioned blow-up issue (that is not present for $r = 0$ and $r = 1$ of course). The proofs utilize some results about ODEs depending on parameters

and an abstract theorem about the convergence of one-step methods, stated in the needed forms in Appendices A and B.

Section 3 contains the core of this paper. We discuss the formulas of the method based on quadratic Taylor approximation and their adjustments based on the aforementioned tolerance considerations, the quantitative a priori as well as run time aspects of step size control to address the blow-up issue, and discuss in detail the numerical algorithms. The Matlab code of the programs can be found in the online supplement.

Section 4 contains the results of numerical tests of the method, using Matlab, with benchmarks against some Runge–Kutta methods of orders 3 and 4, respectively. We have tested the quadratic Taylor method on some standard equations from population dynamics, in line with our original motivation, as well as other equations. Our results on the tested equations confirm that the method based on quadratic Taylor expansion can fare better on the global error by several orders of magnitude when compared to the tested Runge–Kutta methods.

As was mentioned before, we only consider equidistant time-stepping in this paper. We also do not utilize any extrapolation techniques to further improve our method. There are certainly several avenues of investigation, in parallel to established ones for standard numerical methods, that could be pursued to augment the method presented in this paper and improve it further. However, the fact that general Riccati equations do not allow for closed solution formulas is going to remain a limiting factor.

2. Convergence of explicit methods based on exactly solving Taylor approximations of the differential equation

Let $r \in \mathbb{N}_0$, and let $f : D \to \mathbb{R}$ be $(r+1)$-times continuously differentiable on the open set $D \subset \mathbb{R}$, and suppose $y : [0, T] \to D$ solves the initial value problem

$$\begin{cases} \dot{y}(t) = f(y(t)) & \text{on } 0 \le t \le T, \\ y|_{t=0} = y_0 \in D. \end{cases} \tag{2.1}$$

As mentioned in the Introduction, an idea for an explicit method is to locally replace f by its r-th order Taylor polynomial and take the exact solution of the resulting differential equation with the Taylor polynomial instead of f as numerical approximation for $y : [0, T] \to D$ over small time steps. To pursue this idea, define $F : \mathbb{R} \times D \to \mathbb{R}$ via

$$F(w, y) = \sum_{j=0}^{r} \frac{f^{(j)}(y)}{j!} w^j, \tag{2.2}$$

and let $w(h, y)$ for $(h, y) \in U_{\max}$ be the maximally extended solution of

$$\begin{cases} \dfrac{\partial w}{\partial h}(h, y) = F(w(h, y), y), \\ w(0, y) = 0. \end{cases}$$

This differential equation for w depends on y as a parameter, and we have summarized some results about differential equations depending on parameters that we will use below in Appendix B. Since F and all its partial w-derivatives are continuously differentiable with respect to (w, y) in $\mathbb{R} \times D$, we obtain that $\partial_h^k w$ is continuously differentiable with respect to $(h, y) \in U_{\max}$ for all $k \in \mathbb{N}_0$. Define $\Phi : U_{\max} \to \mathbb{R}$ via

$$\Phi(h, y) = y + w(h, y). \tag{2.3}$$

Observe that Φ solves

$$\begin{cases} \dfrac{\partial \Phi}{\partial h}(h, y) = \displaystyle\sum_{j=0}^{r} \frac{f^{(j)}(y)}{j!} (\Phi(h, y) - y)^j, \\ \Phi(0, y) = y. \end{cases}$$

Proposition 2.4. *Φ and all its partial h-derivatives are continuously differentiable with respect to $(h, y) \in U_{\max}$. For every compact subset $K \Subset D$ there exists $h_0 > 0$ such that $\Phi : [0, h_0] \times K \to \mathbb{R}$ is defined, and $\partial\Phi/\partial h : [0, h_0] \times K \to \mathbb{R}$ satisfies a Lipschitz condition with respect to y in $[0, h_0] \times K$.*

Proof. By Theorem B.2 and Remark B.4, w and all its partial h-derivatives exist and are continuously differentiable on $U_{\max}$, and for every $K \Subset D$ there exists $h_0 > 0$ such that $w : [0, h_0] \times K \to \mathbb{R}$ is defined. All this is therefore also true for Φ. Since $\partial^2\Phi/(\partial y\, \partial h) : U_{\max} \to \mathbb{R}$ exists and is continuous, $\partial\Phi/\partial h : [0, h_0] \times K \to \mathbb{R}$ satisfies a Lipschitz condition with respect to y as claimed. □

Proposition 2.5 (local truncation error). *For any compact neighborhood $K \Subset D$ with $y([0, T]) \subset \mathring{K}$ there exist $h_0 > 0$ such that $\Phi : [0, h_0] \times K \to \mathbb{R}$ is defined, and a constant $C \geq 0$ independent of $0 \leq h \leq h_0$ and $0 \leq t \leq T$ such that*

$$|y(t + h) - \Phi(h, y(t))| \leq Ch^{r+2}$$

whenever $0 \leq t + h \leq T$.

If f is a polynomial of degree $\leq r$ we have $y(t+h) = \Phi(h, y(t))$; i.e., the method is locally exact.

Proof. Recall that if u and v are n-times differentiable, $n \in \mathbb{N}$, Faà di Bruno's formula asserts that

$$\frac{d^n}{dh^n}(u \circ v)(h) = \sum_{k=1}^{n} u^{(k)}(v(h)) B_{n,k}(v'(h), v''(h), \ldots, v^{(\mu_k^n)}(h))$$

with the partial Bell polynomials

$$B_{n,k}(x_1,\dots,x_{\mu_k^n}) = \sum_{\substack{\alpha\in\mathbb{N}_0^{\mu_k^n},\ |\alpha|=k\\ 1\alpha_1+2\alpha_2+\cdots+\mu_k^n\alpha_{\mu_k^n}=n}} \frac{n!}{\alpha!}\cdot\Big(\frac{x_1}{1!}\Big)^{\alpha_1}\Big(\frac{x_2}{2!}\Big)^{\alpha_2}\cdots\Big(\frac{x_{\mu_k^n}}{\mu_k^n!}\Big)^{\alpha_{\mu_k^n}},$$

where $\mu_k^n = n-k+1$. We now proceed to use this formula in order to show inductively that

$$\frac{d^n}{dh^n}y(t+h)\Big|_{h=0} = \frac{\partial^n}{\partial h^n}\Phi(h,y(t))\Big|_{h=0} \tag{2.6}$$

for $n=0,\dots,r+1$ (note that $y\in C^{r+2}([0,T])$ since $f\in C^{r+1}(D)$ by assumption). For $n=0$ this follows immediately from the definition in (2.3), keeping in mind that $w(0,y)=0$. For $n=1$ we have

$$\begin{aligned}\frac{d}{dh}y(t+h)\Big|_{h=0} &= f(y(t+h))|_{h=0} = f(y(t)),\\ \frac{\partial}{\partial h}\Phi(h,y(t))\Big|_{h=0} &= F(w(h,y(t)),y(t))|_{h=0} = F(0,y(t)) = f(y(t)).\end{aligned}$$

So suppose we know (2.6) for all $n\le n_0$ for some $1\le n_0\le r$. Now

$$\begin{aligned}\frac{d^{n_0+1}}{dh^{n_0+1}}y(t+h) &= \frac{d^{n_0}}{dh^{n_0}}\Big(\frac{d}{dh}y(t+h)\Big) = \frac{d^{n_0}}{dh^{n_0}}(f\circ y)(t+h)\\ &= \sum_{k=1}^{n_0} f^{(k)}(y(t+h))B_{n_0,k}(y'(t+h),y''(t+h),\dots,y^{(\mu_k^{n_0})}(t+h)).\end{aligned}$$

Evaluation at $h=0$ gives

$$\frac{d^{n_0+1}}{dh^{n_0+1}}y(t+h)\Big|_{h=0} = \sum_{k=1}^{n_0} f^{(k)}(y(t))B_{n_0,k}(y'(t),y''(t),\dots,y^{(\mu_k^{n_0})}(t)). \tag{2.7}$$

Using the differential equation for $w(h,y)$ and (2.3) we get

$$\frac{\partial^{n_0+1}}{\partial h^{n_0+1}}\Phi(h,y(t))\Big|_{h=0} = \frac{\partial^{n_0}}{\partial h^{n_0}}F(w(h,y(t)),y(t))\Big|_{h=0},$$

which by Faà di Bruno's formula equals

$$\sum_{k=1}^{n_0}(\partial_w^k F)(w(0,y(t)),y(t))\,B_{n_0,k}\big((\partial_h w)(0,y(t)),\dots,(\partial_h^{\mu_k^{n_0}}w)(0,y(t))\big). \tag{2.8}$$

By induction, $(\partial_h^j w)(0,y(t)) = y^{(j)}(t)$ for $j=1,\dots,\mu_k^{n_0}$, and thus the arguments in the partial Bell polynomials $B_{n_0,k}$ in (2.7) and (2.8) agree. Moreover,

$$(\partial_w^k F)(w(0,y(t)),y(t)) = (\partial_w^k F)(0,y(t)) = f^{(k)}(y(t))$$

for $k=0,\dots,r$ in view of (2.2) and Taylor's formula. This shows that (2.6) holds for $n=n_0+1$ and finishes the induction.

In view of (2.6), Taylor's formula now implies

$$|y(t+h)-\Phi(h,y(t))| = \left| \int_0^h \frac{y^{(r+2)}(t+s)-(\partial_h^{r+2}\Phi)(s,y(t))}{(r+1)!}(h-s)^{r+1}\,ds \right|$$
$$\leq \underbrace{\left[\frac{1}{(r+2)!}\Big(\max_{0\leq s\leq T} |y^{(r+2)}(s)| + \max_{\substack{0\leq s\leq h_0 \\ y\in K}} |(\partial_h^{r+2}\Phi)(s,y)| \Big) \right]}_{=:C} \cdot h^{r+2}$$

for all $0 \leq t \leq T$ and $0 \leq h \leq h_0$ such that $t+h \leq T$.

If f is a polynomial of degree $\leq r$ we have

$$f(z) = \sum_{j=0}^{r} \frac{f^{(j)}(y)}{j!}(z-y)^j$$

for all $z, y \in \mathbb{R}$, and consequently both $h \mapsto y(t+h)$ and $h \mapsto \Phi(h, y(t))$ solve

$$\begin{cases} \dot{u}(h) = f(u(h)), & h \geq 0, \\ u|_{h=0} = y(t). \end{cases}$$

By uniqueness we must therefore have $y(t+h) = \Phi(h, y(t))$. □

Theorem 2.9. *The method* Φ *defined in* (2.3) *is a convergent method of order* $r+1$ *for the approximation of the solution* $y : [0, T] \to \mathbb{R}$ *of* (2.1). *The method is exact for differential equations* (2.1) *when* f *is a polynomial of degree at most* r.

Proof. This follows with Propositions 2.4 and 2.5 from Theorem A.1. □

Example 2.10. If we specialize to $r = 0$ and $r = 1$ in (2.2) we find familiar methods.

- If $r = 0$ we have $F(w, h) = f(y)$ in (2.2), and so $\Phi(h, y)$ solves

$$\begin{cases} \dfrac{\partial \Phi}{\partial h}(h, y) = f(y), \\ \Phi(0, y) = y. \end{cases}$$

Thus $\Phi(h, y) = y + hf(y)$ is the Euler method.

- If $r = 1$ we have $F(w, y) = f(y) + f'(y)w$ in (2.2), and so $\Phi(h, y)$ solves

$$\begin{cases} \dfrac{\partial \Phi}{\partial h}(h, y) = f(y) + f'(y)(\Phi(h, y) - y), \\ \Phi(0, y) = y. \end{cases}$$

Thus $\Phi(h, y) = y + h\phi_1(f'(y)h)f(y)$ with

$$\phi_1(z) = \begin{cases} (e^z - 1)/z & \text{for } z \neq 0, \\ 1 & \text{for } z = 0 \end{cases}$$

is the Rosenbrock–Euler method [Hochbruck and Ostermann 2010, Section 2.4].

In this paper, we present and analyze the method based on adaptive Taylor approximation in detail for $r = 2$. While the theoretical result in Theorem 2.9 holds for all $r \in \mathbb{N}_0$, it is not feasible for the implementation of methods for larger r as one generally does not have explicit solution formulas for polynomial ordinary differential equations.

3. Third-order scheme based on quadratic Taylor approximation

We begin by defining and analyzing the analytic function

$$\psi(u, v) = \frac{\sinh(v)}{u\sinh(v) + v\cosh(v)} \tag{3.1}$$

depending on two complex variables $(u, v) \in \mathbb{C}^2$. Initially, this function is undefined on

$$S = \{(u, v) \in \mathbb{C}^2 : u\sinh(v) + v\cosh(v) = 0\}.$$

It is easy to see that S consists of the complex line $v = 0$ and the complex surface

$$S_\psi : u = -v\coth(v).$$

The singularities of ψ where $v = 0$ are removable, except for the singularity at the single branch point $(-1, 0)$. To see this note that for v near 0 we can write

$$\psi(u, v) = \frac{1}{u + v\coth(v)}.$$

The function $v \mapsto v\coth(v)$ has a removable singularity at $v = 0$. The first few terms of the Taylor series are

$$v\coth(v) = 1 + \frac{v^2}{3} - \frac{v^4}{45} + O(v^6),$$

which shows that ψ has a removable singularity at all points $(u, 0)$ if $u \neq -1$. More precisely, we get

$$\psi(u, v) = \frac{1}{1+u} - \frac{v^2}{3(1+u)^2} + \frac{(u+6)v^4}{45(1+u)^3} + O(v^6) \tag{3.2}$$

locally uniformly in u, and, in particular, we see that the definition

$$\psi(u, 0) = \frac{1}{1+u}, \quad u \neq -1, \tag{3.3}$$

extends ψ analytically to $v = 0$ except the branch point. From (3.1) and (3.3) we obtain that ψ remains singular on $S_\psi : u = -v\coth(v)$, but now with the understanding that the singularity of $v\coth(v)$ when $v = 0$ has been removed. We are going to need the function ψ only in the cases that both u and v are real, or that

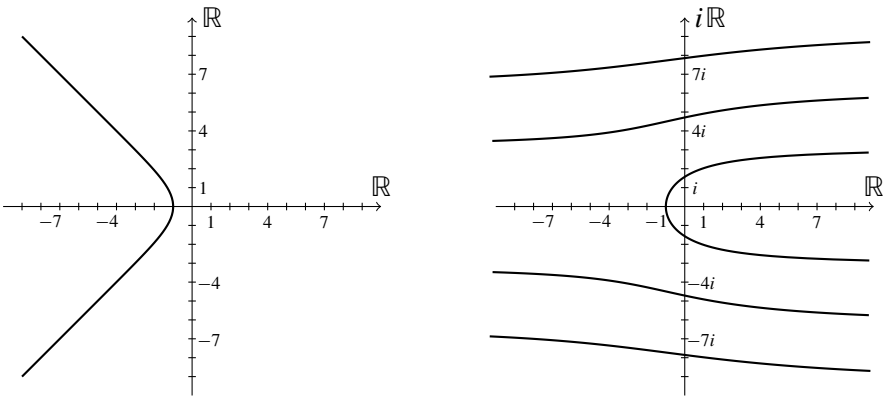

Figure 1. Singularities of ψ: $S_\psi \cap \mathbb{R}^2$ (left) and $S_\psi \cap (\mathbb{R} \times i\mathbb{R})$ (right).

u is real and v is imaginary. Figure 1 shows parts of the intersection of the singular set S_ψ with $\mathbb{R}^2$ and with $\mathbb{R} \times i\mathbb{R}$, respectively.

The relevance of the function ψ for us is clarified by the following lemma. The proof is straightforward and will be omitted.

Lemma 3.4. *Consider the initial value problem for the Riccati ordinary differential equation*

$$\begin{cases} \dot{w} = aw^2 + bw + c, \\ w|_{t=0} = 0, \end{cases} \tag{3.5}$$

with constant coefficients $a, b, c \in \mathbb{R}$. Let $\Delta = b^2 - 4ac$, and define $\alpha = -\frac{1}{2}b \in \mathbb{R}$, $\beta = \frac{1}{2}\sqrt{\Delta} \in \mathbb{C}$. Note that $\beta \in \mathbb{R}$ if $\Delta \geq 0$, and in the case $\Delta < 0$ we choose $\sqrt{\Delta}$ to be the root with positive imaginary part,[1] *so $\beta \in i\mathbb{R}_+$.*

The maximal solution to (3.5) *is given by*

$$w(t) \equiv w(t; a, b, c) = ct\psi(t\alpha, t\beta), \quad t_{\min} < t < t_{\max},$$

where

$$t_{\min} = \sup\{t < 0 : (t\alpha, t\beta) \in S_\psi\} \in \mathbb{R}_- \cup \{-\infty\},$$
$$t_{\max} = \inf\{t > 0 : (t\alpha, t\beta) \in S_\psi\} \in \mathbb{R}_+ \cup \{\infty\}.$$

It is important to note that the solution $w(t)$ to (3.5) can blow up in finite time depending on the values of the constants a, b, c. This has a serious impact on the method presented here in that additional integrity checks on the step size must be performed (a priori or at run time) that do not appear in Runge–Kutta methods or exponential integrators. The way the solution $w(t)$ is represented in Lemma 3.4 utilizing the function ψ facilitates a simple visualization of the existence interval. As

[1] We could choose either, really, since $\psi(u, v)$ is even in v.

described in the lemma, the coefficients $a, b, c \in \mathbb{R}$ determine a point $(\alpha, \beta) \in \mathbb{R}^2$, or $(\alpha, \beta) \in \mathbb{R} \times i\mathbb{R}$, respectively, and the solution involves evaluation of the function ψ restricted to the line passing through the origin in $\mathbb{R}^2$ (or $\mathbb{R} \times i\mathbb{R}$) and that point, parametrized by t. Then $t_{\max}$ is precisely the first positive t-value when that line crosses the singular set S_ψ of the function ψ (and $t_{\max} = \infty$ if there is no such crossing point), and similarly for $t_{\min}$. It is easy to visualize this behavior in Figure 1. We summarize, focusing on $t_{\max}$:

- $\Delta \geq 0$, so $(\alpha, \beta) \in \mathbb{R}^2$: $t_{\max} < \infty$ precisely when $\alpha < 0$ and $|\beta| < |\alpha|$. These conditions are equivalent to $b > 0$ and $0 < ac \leq \frac{1}{4}b^2$. By the definition of S_ψ, $t_{\max}$ is then the (unique) solution to

$$\alpha t_{\max} = -\beta t_{\max} \coth(\beta t_{\max})$$

(recall that $v \coth(v) = 1$ when $v = 0$). Consequently,

$$t_{\max} = \begin{cases} -\dfrac{1}{\alpha} = \dfrac{2}{b} & \text{if } \Delta = 0, \\ \dfrac{1}{\beta} \operatorname{arcoth}\Big(-\dfrac{\alpha}{\beta}\Big) = \dfrac{1}{\sqrt{\Delta}} \ln\Big(\dfrac{b+\sqrt{\Delta}}{b-\sqrt{\Delta}}\Big) & \text{if } \Delta > 0. \end{cases}$$

- If $\Delta < 0$ we have $t_{\max} < \infty$, and

$$\alpha t_{\max} = -\beta t_{\max} \coth(\beta t_{\max}) = -(-i\beta) t_{\max} \cot(-i\beta t_{\max}).$$

We get[2]

$$t_{\max} = \frac{1}{(-i\beta)} \operatorname{arccot}\Big(-\frac{\alpha}{(-i\beta)}\Big) = \frac{2}{\sqrt{-\Delta}} \operatorname{arccot}\Big(\frac{b}{\sqrt{-\Delta}}\Big).$$

In summary,

$$t_{\max} = \begin{cases} \dfrac{2}{b} & \text{if } \Delta = 0,\ b > 0, \\ \dfrac{1}{\sqrt{\Delta}} \ln\Big(\dfrac{b+\sqrt{\Delta}}{b-\sqrt{\Delta}}\Big) & \text{if } \Delta > 0,\ \sqrt{\Delta} < b, \\ \dfrac{2}{\sqrt{-\Delta}} \operatorname{arccot}\Big(\dfrac{b}{\sqrt{-\Delta}}\Big) & \text{if } \Delta < 0, \\ \infty & \text{otherwise.} \end{cases}$$

In either case, since the closest point of $S_\psi \cap \mathbb{R}^2$, or $S_\psi \cap (\mathbb{R} \times i\mathbb{R})$, respectively, to the origin with respect to the Euclidean distance $|(\cdot, \cdot)|$ is the point $(-1, 0)$, we note that the solution is guaranteed to exist while $|(t\alpha, t\beta)| < 1$. This gives a rough estimate,

$$t_{\max} \geq \frac{1}{|(\alpha, \beta)|} = \frac{2}{\sqrt{b^2 + |\Delta|}}, \tag{3.6}$$

[2]In the formula for $t_{\max}$ and elsewhere we use the real $\operatorname{arccot} : \mathbb{R} \to (0, \pi)$.

where we understand the right-hand side of (3.6) to be ∞ if $b = \Delta = 0$. This estimate can be used to derive an a priori estimate on valid step sizes of the method, as described below.

Description of the method. The goal is to approximate the solution $y = y(t)$ to the initial value problem

$$\begin{cases} \dot{y} = f(y), \\ y|_{t=0} = y_0, \end{cases}$$

for $(t, y) \in [0, T] \times [A, B]$. We assume that f is C^3 in an open neighborhood of $[A, B]$, and $y_0 \in [A, B]$. To fix notation, define functions $a, b, c, \Delta, h_{\max} : [A, B] \to \mathbb{R} \cup \{\infty\}$ via

$$a(y) = \frac{f''(y)}{2}, \quad b(y) = f'(y), \quad c(y) = f(y), \quad \Delta = b^2 - 4ac,$$

$$h_{\max}(y) = \begin{cases} \dfrac{2}{b(y)} & \text{if } \Delta(y) = 0,\ b(y) > 0, \\ \dfrac{1}{\sqrt{\Delta(y)}} \ln\left(\dfrac{b(y) + \sqrt{\Delta(y)}}{b(y) - \sqrt{\Delta(y)}}\right) & \text{if } \Delta(y) > 0,\ \sqrt{\Delta(y)} < b(y), \\ \dfrac{2}{\sqrt{-\Delta(y)}} \operatorname{arccot}\left(\dfrac{b(y)}{\sqrt{-\Delta(y)}}\right) & \text{if } \Delta(y) < 0, \\ \infty & \text{otherwise.} \end{cases}$$

Choose a tolerance $0 < \mathrm{tol}_0 \ll 1$. Quantities that in absolute value are less than tol_0 are considered numerically zero. Evaluation of approximate $0/0$ expressions with denominators of magnitude $< \mathrm{tol}_0$, such as occur in the evaluation of $\psi(u, v)$ given by (3.1) for v near zero, should be avoided to improve stability. For this reason, we define

$$\Phi(h, y) = \begin{cases} y + \dfrac{2c(y)\sinh\left[\frac{1}{2}\sqrt{\Delta(y)}h\right]}{\sqrt{\Delta(y)}\cosh\left[\frac{1}{2}\sqrt{\Delta(y)}h\right] - b(y)\sinh\left[\frac{1}{2}\sqrt{\Delta(y)}h\right]} & \text{for } (h, y) \in U_+, \\ y + \dfrac{2c(y)\sin\left[\frac{1}{2}\sqrt{-\Delta(y)}h\right]}{\sqrt{-\Delta(y)}\cos\left[\frac{1}{2}\sqrt{-\Delta(y)}h\right] - b(y)\sin\left[\frac{1}{2}\sqrt{-\Delta(y)}h\right]} & \text{for } (h, y) \in U_-, \\ y + \dfrac{2c(y)h}{2 - b(y)h} - \dfrac{h^3 c(y)\Delta(y)}{3(2 - b(y)h)^2} & \text{for } (h, y) \in U_0, \end{cases} \tag{3.7}$$

where

$$U_+ = \{(h, y) \in [0, \infty) \times [A, B] : \Delta(y) \geq 4\mathrm{tol}_0,\ h < h_{\max}(y),\ 2 - hb(y) \geq \sqrt{\mathrm{tol}_0}\},$$

$$U_- = \{(h, y) \in [0, \infty) \times [A, B] : \Delta(y) \leq -4\mathrm{tol}_0,\ h < h_{\max}(y),\ 2 - hb(y) \geq \sqrt{\mathrm{tol}_0}\},$$

$$U_0 = \{(h, y) \in [0, \infty) \times [A, B] : |\Delta(y)| < 4\mathrm{tol}_0,\ 2 - hb(y) \geq \sqrt{\mathrm{tol}_0}\}.$$

Some comments are in order:

• By Lemma 3.4 and formula (3.1), the first two cases in (3.7) are the exact formulas of the general method (2.3) discussed in Section 2 with $r = 2$. In the second case we merely converted to trigonometric functions in the formulas since the argument of the hyperbolic trigonometric functions would be imaginary.

• Taylor expansion of the hyperbolic trigonometric functions in the denominator in the first case gives

$$\begin{aligned}
&\sqrt{\Delta(y)}\cosh\big[\tfrac{1}{2}\sqrt{\Delta(y)}h\big] - b(y)\sinh\big[\tfrac{1}{2}\sqrt{\Delta(y)}h\big] \\
&\qquad = \sqrt{\Delta(y)}\sum_{k=0}^{\infty}\frac{\big[\tfrac{1}{2}\sqrt{\Delta(y)}h\big]^{2k}}{(2k)!} - b(y)\sum_{k=0}^{\infty}\frac{\big[\tfrac{1}{2}\sqrt{\Delta(y)}h\big]^{2k+1}}{(2k+1)!} \\
&\qquad = \frac{\sqrt{\Delta(y)}}{2}\sum_{k=0}^{\infty}\frac{1}{(2k)!}\left[2 - \frac{hb(y)}{2k+1}\right]\big[\tfrac{1}{2}\sqrt{\Delta(y)}h\big]^{2k} \\
&\qquad \geq \mathrm{tol}_0\cdot\cosh\big[\tfrac{1}{2}\sqrt{\Delta(y)}h\big] \geq \mathrm{tol}_0
\end{aligned}$$

under the assumption that both $\frac{1}{2}\sqrt{\Delta(y)} \geq \sqrt{\mathrm{tol}_0}$ and $2 - hb(y) \geq \sqrt{\mathrm{tol}_0}$, which explains the definition of U_+.

• In the second case, we note that when $b(y) > 0$ we have $h_{\max}(y) < 2/b(y)$ directly from the definition when $\Delta(y) < 0$, and consequently $2 - hb(y) > 0$ is implied by $h < h_{\max}(y)$ (the inequality is trivially fulfilled for $b(y) \leq 0$). The condition $2 - hb(y) \geq \sqrt{\mathrm{tol}_0}$ gives an extra buffer. We also note, arguing analogous to the first case, that

$$\begin{aligned}
&\sqrt{-\Delta(y)}\cos\big[\tfrac{1}{2}\sqrt{-\Delta(y)}h\big] - b(y)\sin\big[\tfrac{1}{2}\sqrt{-\Delta(y)}h\big] \\
&\qquad = \tfrac{1}{2}\sqrt{-\Delta(y)}\sum_{k=0}^{\infty}\frac{(-1)^k}{(2k)!}\left[2 - \frac{hb(y)}{2k+1}\right]\big[\tfrac{1}{2}\sqrt{-\Delta(y)}h\big]^{2k} \\
&\qquad = \tfrac{1}{2}\sqrt{-\Delta(y)}\big[2 - hb(y) + O\big[\big[\tfrac{1}{2}\sqrt{-\Delta(y)}h\big]^2\big]\big],
\end{aligned}$$

and thus the denominator is asymptotically $\geq \mathrm{tol}_0$ under the restrictions placed on U_-.

• If the discriminant term $\Delta(y)$ is too small, evaluation of $\psi(u, v)$ as given by (3.1) is unstable, so we opt to use the expansion (3.2) instead for such terms, leading to the definition of $\Phi(h, y)$ in the third case. By Lemma 3.4 and expansion (3.2), we note that the theoretical method as determined by (2.3) and our definition for $\Phi(h, y)$ in the third case of (3.7) coincide to third order in h as $h \to 0$, showing that the local truncation error in our definition is still $O(h^4)$ as required. Moreover, our definition for $\Phi(h, y)$ in the third case matches the general method from (2.3) if $\Delta(y) = 0$.

The algorithm. Besides the differential equation $\dot{y} = f(y)$ and the initial value y_0, the inputs are $0 < \mathrm{tol}_0 \ll 1$, the window $[0, T] \times [A, B]$ where the solution $y(t)$ is supposed to be approximated, and the chosen step size $h > 0$ for constructing an approximating sequence $y_0, y_1, \ldots$ of values for the solution at equidistant points $t = jh$, $j = 0, 1, \ldots$.

(1) Check whether $y_0 \in [A, B]$. If not, the algorithm terminates with an error message that the initial value lies outside of the chosen tracking window.

Now suppose that an approximating partial sequence $y_0, \ldots, y_n$ for some $n \in \mathbb{N}_0$ has already been successfully constructed.

(2) If $(n + 1)h > T$, the algorithm terminates with success and displays the approximation $y_0, \ldots, y_n$ of the solution.

(3) *Integrity check on the step size*: Check whether $(h, y_n) \in U_+ \cup U_- \cup U_0$. If not, the program terminates with the message that the algorithm stops after n steps, approximating the solution on $[0, nh]$, as the method becomes undefined in the next step due to the chosen step size. The approximation of the solution thus far is displayed, and it is suggested to run the program again with a smaller step size $h > 0$.

(4) Check whether $\Phi(h, y_n) \in [A, B]$. If not, the program is terminated with the message that the algorithm stops after n steps, approximating the solution on $[0, nh]$, as the approximate solution is leaving the designated tracking window in the next step. The approximation of the solution thus far is displayed.

(5) If the program reaches this step, it accepts $y_{n+1} = \Phi(h, y_n)$ as the next value of the approximating sequence, and recursively resumes at step (2) with n incremented by one.

Instead of performing the integrity check on the step size in (3) at run time during every execution of the recursive loop, an a priori estimate can be obtained prior to building the approximating sequence to determine a value $h_0 > 0$ that only depends on f, tol_0, and the chosen viewing window such that all step sizes $0 < h < h_0$ work. Following this procedure and skipping the integrity checks at run time increases the speed of the program. The a priori estimate utilizes (3.6), as follows:

(i) Find the maximum value $b_{\max}$ of $b : [A, B] \to \mathbb{R}$.

(ii) Find the maximum value $s_{\max}$ of $s := b^2 + |\Delta| : [A, B] \to \mathbb{R}$.

(iii) Set

$$h_0 = \begin{cases} \min\{2/\sqrt{s_{\max}}, (2 - \sqrt{\mathrm{tol}_0})/b_{\max}, T\} & \text{if } s_{\max} > \mathrm{tol}_0,\ b_{\max} > \mathrm{tol}_0, \\ \min\{2/\sqrt{s_{\max}}, T\} & \text{if } s_{\max} > \mathrm{tol}_0,\ b_{\max} \le \mathrm{tol}_0, \\ T & \text{otherwise.} \end{cases}$$

4. Numerical tests of the quadratic Taylor method

In all tests described below we used the tolerance $\mathrm{tol}_0 = 1 \cdot 10^{-14}$ and recorded the global error and the run time of the method on the indicated interval for the problem with various step sizes h. Errors in magnitude less than tol_0 have been recorded as zero. All tests were performed using Matlab (Version 2018a, Update 4) on a ThinkPad laptop computer equipped with an Intel i7-7500U CPU @ 2.70GHz and 16.0 GB RAM, running Windows 10. We are benchmarking our third-order method, labeled QT3 below, against the following standard methods from the Runge–Kutta family:

• Kutta third-order method (K3), see [Butcher 2008, Section 233]: the Butcher tableau for this method is

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

• Bogacki–Shampine third-order method (BS3) [1989]: the Butcher tableau for this method is

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{3}{4} & 0 & \frac{3}{4} & & \\ 1 & \frac{2}{9} & \frac{1}{3} & \frac{4}{9} & \\ \hline & \frac{2}{9} & \frac{1}{3} & \frac{4}{9} & 0 \end{array}$$

Embedded in a 3(2) pair, this method is built into one of the standard algorithms, `ode23`, of the Matlab suite [Shampine and Reichelt 1997]. In [Bogacki and Shampine 1989] the third-order formulas are credited to [Ralston 1965].

• Classical Runge–Kutta fourth-order method (RK4), see [Hairer et al. 1993, Section II.1]: the Butcher tableau for this method is

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

Logistic equation. As expected, the quadratic Taylor method outperforms standard Runge–Kutta methods for quadratic ordinary differential equations with regards to

h	K3	BS3	RK4	QT3
0.1	$9.0574 \cdot 10^{-2}$	$4.9747 \cdot 10^{-2}$	$1.3532 \cdot 10^{-2}$	0
0.05	$1.3495 \cdot 10^{-2}$	$8.2625 \cdot 10^{-3}$	$1.0941 \cdot 10^{-3}$	0
0.02	$9.6842 \cdot 10^{-4}$	$6.3000 \cdot 10^{-4}$	$3.3012 \cdot 10^{-5}$	0
0.01	$1.2579 \cdot 10^{-4}$	$8.3520 \cdot 10^{-5}$	$2.1834 \cdot 10^{-6}$	0

Table 1. Global error vs. step size for (4.1).

h	K3	BS3	RK4	QT3
0.1	0.396497	0.507937	0.498741	0.823234
0.05	0.652171	0.860344	0.896938	1.625599
0.02	1.681197	2.138214	2.149054	3.842002
0.01	3.413213	4.193556	4.182386	7.75049

Table 2. Run time (in seconds) vs. step size for (4.1).

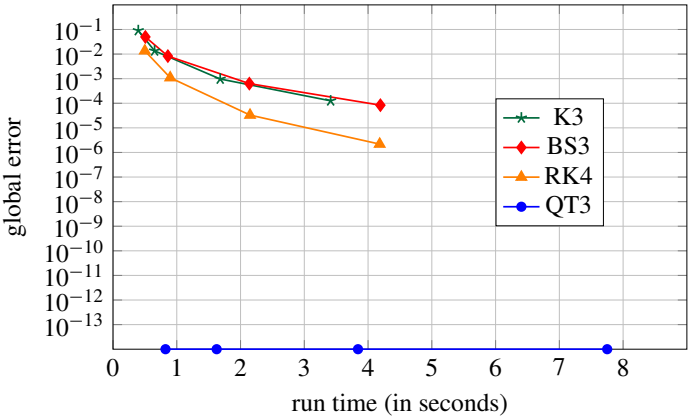

Figure 2. Work-precision diagram for (4.1) from Tables 1 and 2.

accuracy. Consider

$$\begin{cases} \dot{y} = y(10 - y), \\ y|_{t=0} = 0.5, \end{cases} \qquad 0 \le t \le 2. \tag{4.1}$$

The exact solution is

$$y(t) = \frac{10e^{10t}}{19 + e^{10t}}, \quad 0 \le t \le 2.$$

The results are displayed in Tables 1 and 2, and in Figures 2 and 3.

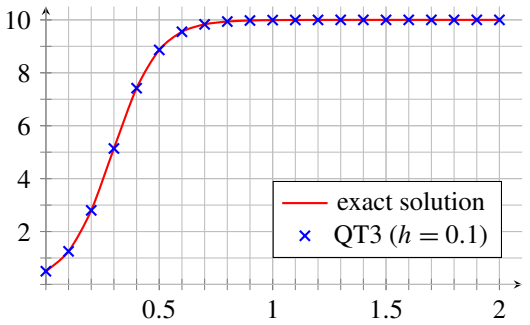

Figure 3. Exact solution and QT3 values for (4.1).

h	K3	BS3	RK4	QT3
0.1	$2.8543 \cdot 10^{-6}$	$2.8543 \cdot 10^{-6}$	$5.6900 \cdot 10^{-8}$	$9.6127 \cdot 10^{-13}$
0.05	$3.7135 \cdot 10^{-7}$	$3.7135 \cdot 10^{-7}$	$3.7073 \cdot 10^{-9}$	$1.2390 \cdot 10^{-13}$
0.02	$2.4343 \cdot 10^{-8}$	$2.4343 \cdot 10^{-8}$	$9.7307 \cdot 10^{-11}$	0
0.01	$3.0673 \cdot 10^{-9}$	$3.0673 \cdot 10^{-9}$	$6.1326 \cdot 10^{-12}$	0

Table 3. Global error vs. step size for (4.2).

h	K3	BS3	RK4	QT3
0.1	0.862458	1.125509	1.131106	2.028254
0.05	1.622196	2.122481	2.171911	3.874084
0.02	4.218344	5.616584	5.505497	10.525293
0.01	8.255611	11.230867	11.15308	20.897061

Table 4. Run time (in seconds) vs. step size for (4.2).

Bernoulli equation. Consider

$$\begin{cases} \dot{y} = y\big(1 - \big(\frac{1}{20}y\big)^2\big), \\ y|_{t=0} = 1 \cdot 10^{-4}, \end{cases} \qquad 0 \le t \le 5. \tag{4.2}$$

The exact solution is

$$y(t) = \frac{20}{\sqrt{(4 \cdot 10^{10} - 1)e^{-2t} + 1}}, \quad 0 \le t \le 5.$$

The results are displayed in Tables 3 and 4, and in Figures 4 and 5.

Let's also consider the same differential equation

$$\begin{cases} \dot{y} = y\big(1 - \big(\frac{1}{20}y\big)^2\big), \\ y|_{t=0} = 1, \end{cases} \qquad 0 \le t \le 5, \tag{4.3}$$

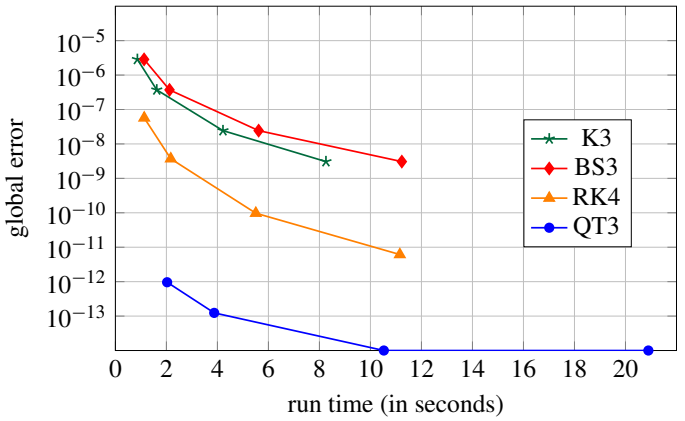

Figure 4. Work-precision diagram for (4.2) from Tables 3 and 4.

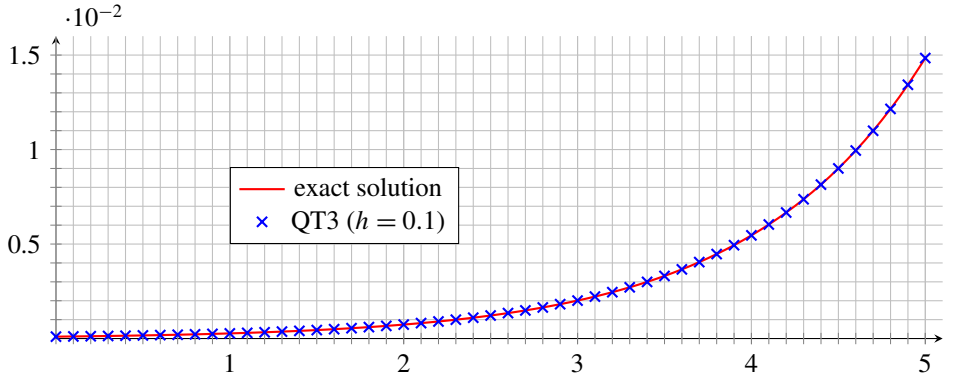

Figure 5. Exact solution and QT3 values for (4.2).

but with a different initial value that is farther away from the equilibrium solutions. The exact solution is then

$$y(t) = \frac{20}{\sqrt{399e^{-2t} + 1}}, \quad 0 \le t \le 5.$$

The results are displayed in Tables 5 and 6, and in Figures 6 and 7.

h	K3	BS3	RK4	QT3
0.1	$6.3817 \cdot 10^{-4}$	$4.5295 \cdot 10^{-4}$	$1.5055 \cdot 10^{-5}$	$3.2525 \cdot 10^{-4}$
0.05	$8.1554 \cdot 10^{-5}$	$5.8683 \cdot 10^{-5}$	$9.2633 \cdot 10^{-7}$	$4.1018 \cdot 10^{-5}$
0.02	$5.2845 \cdot 10^{-6}$	$3.8374 \cdot 10^{-6}$	$2.3554 \cdot 10^{-8}$	$2.6396 \cdot 10^{-6}$
0.01	$6.6341 \cdot 10^{-7}$	$4.8314 \cdot 10^{-7}$	$1.4695 \cdot 10^{-9}$	$3.3052 \cdot 10^{-7}$

Table 5. Global error vs. step size for (4.3).

h	K3	BS3	RK4	QT3
0.1	0.871428	1.135167	1.204247	1.991003
0.05	1.705217	2.190155	2.190203	4.047121
0.02	4.002053	5.512081	5.56601	9.936015
0.01	8.80262	11.240246	13.179334	18.890316

Table 6. Run time (in seconds) vs. step size for (4.3).

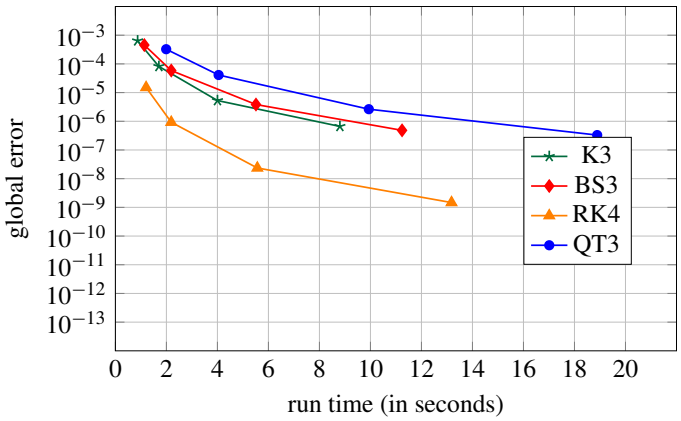

Figure 6. Work-precision diagram for (4.3) from Tables 5 and 6.

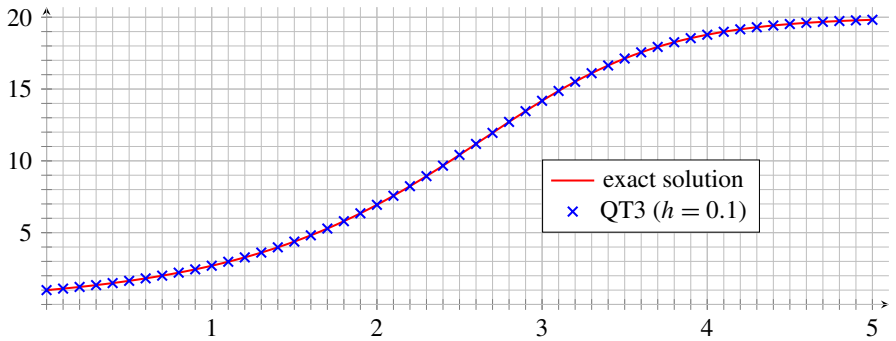

Figure 7. Exact solution and QT3 values for (4.3).

Gompertz equation. Consider

$$\begin{cases} \dot{y} = y \ln(30/y), \\ y|_{t=0} = 29, \end{cases} \qquad 0 \le t \le 2. \tag{4.4}$$

The exact solution is

$$y(t) = 30\left(\tfrac{29}{30}\right)^{e^{-t}}, \quad 0 \le t \le 2.$$

The results are displayed in Tables 7 and 8, and in Figures 8 and 9.

h	K3	BS3	RK4	QT3
0.1	$1.5931 \cdot 10^{-5}$	$1.5604 \cdot 10^{-5}$	$3.1690 \cdot 10^{-7}$	$9.7263 \cdot 10^{-9}$
0.05	$1.9169 \cdot 10^{-6}$	$1.8770 \cdot 10^{-6}$	$1.9019 \cdot 10^{-8}$	$1.1837 \cdot 10^{-9}$
0.02	$1.1990 \cdot 10^{-7}$	$1.1734 \cdot 10^{-7}$	$4.7509 \cdot 10^{-10}$	$7.4419 \cdot 10^{-11}$
0.01	$1.4873 \cdot 10^{-8}$	$1.4554 \cdot 10^{-8}$	$2.9431 \cdot 10^{-11}$	$9.2619 \cdot 10^{-12}$

Table 7. Global error vs. step size for (4.4).

h	K3	BS3	RK4	QT3
0.1	0.389789	0.499958	0.483573	0.801802
0.05	0.678447	0.890931	0.892354	1.570274
0.02	1.572932	2.228926	2.24712	4.003964
0.01	3.329032	4.399356	4.406119	8.440522

Table 8. Run time (in seconds) vs. step size for (4.4).

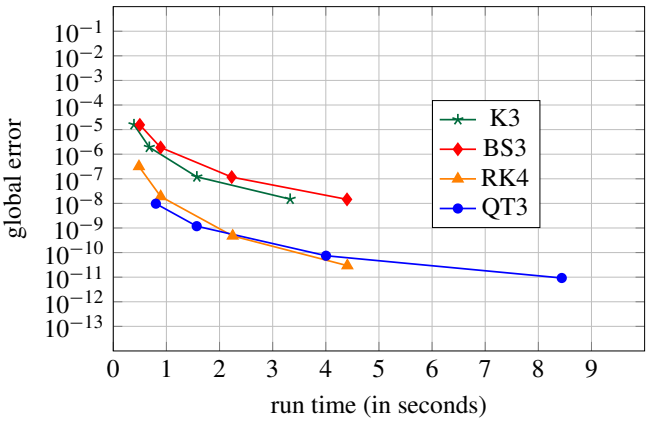

Figure 8. Work-precision diagram for (4.4) from Tables 7 and 8.

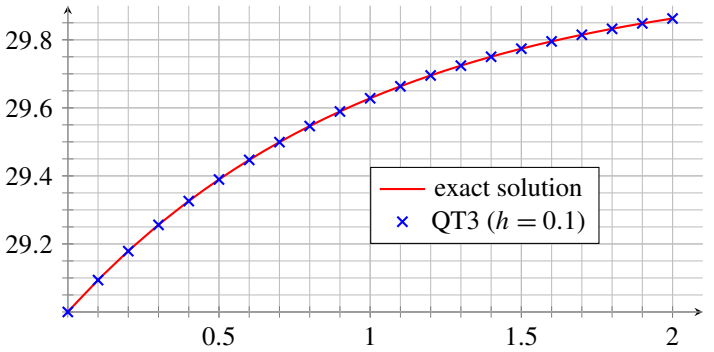

Figure 9. Exact solution and QT3 values for (4.4).

h	K3	BS3	RK4	QT3
0.1	$3.0134 \cdot 10^{-7}$	$2.8743 \cdot 10^{-7}$	$5.9219 \cdot 10^{-9}$	$3.8462 \cdot 10^{-10}$
0.05	$3.6318 \cdot 10^{-8}$	$3.4589 \cdot 10^{-8}$	$3.5555 \cdot 10^{-10}$	$4.6768 \cdot 10^{-11}$
0.02	$2.2745 \cdot 10^{-9}$	$2.1638 \cdot 10^{-9}$	$8.8861 \cdot 10^{-12}$	$2.9453 \cdot 10^{-12}$
0.01	$2.8224 \cdot 10^{-10}$	$2.6843 \cdot 10^{-10}$	$5.5067 \cdot 10^{-13}$	$3.6637 \cdot 10^{-13}$

Table 9. Global error vs. step size for (4.5).

h	K3	BS3	RK4	QT3
0.1	1.635349	2.132353	2.109736	3.99236
0.05	3.471573	4.173892	4.026089	8.406138
0.02	8.569636	10.934895	10.836708	19.228195
0.01	16.240501	20.771019	20.858536	40.610184

Table 10. Run time (in seconds) vs. step size for (4.5).

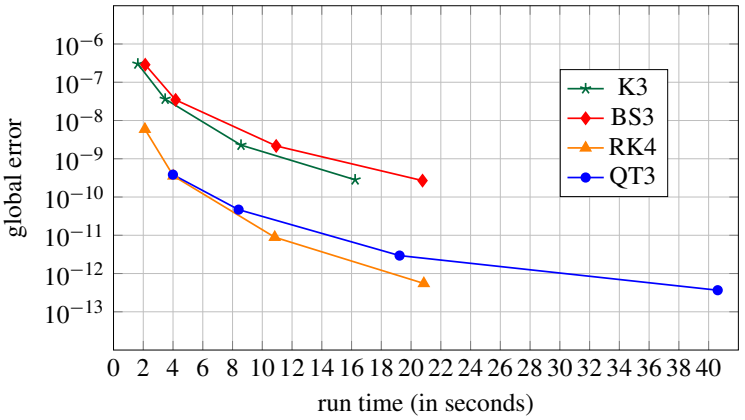

Figure 10. Work-precision diagram for (4.5) from Tables 9 and 10.

Flame propagation. The following example is taken from a *Cleve's Corner* blog post on the MathWorks web page; see [Moler 2003]. It is attributed there to L. Shampine. Consider

$$\begin{cases} \dot{y} = y^2 - y^3, \\ y|_{t=0} = 0.98, \end{cases} \qquad 0 \le t \le 10. \tag{4.5}$$

The exact solution is

$$y(t) = \frac{1}{1 + W\left(\frac{1}{49} e^{1/49 - t}\right)}, \quad 0 \le t \le 10,$$

where W is the Lambert W function; see [Corless et al. 1996]. The results are displayed in Tables 9 and 10, and in Figures 10 and 11.

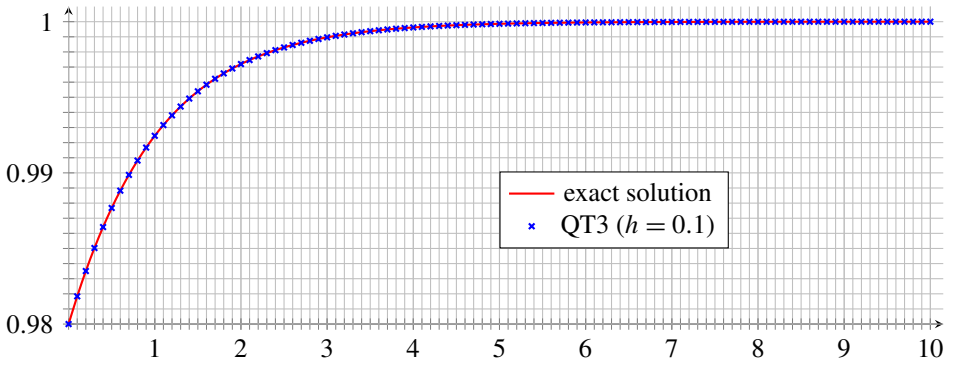

Figure 11. Exact solution and QT3 values for (4.5).

h	K3	BS3	RK4	QT3
0.1	$1.0453 \cdot 10^{-6}$	$1.0450 \cdot 10^{-6}$	$2.0837 \cdot 10^{-8}$	$3.4029 \cdot 10^{-10}$
0.05	$1.3599 \cdot 10^{-7}$	$1.3594 \cdot 10^{-7}$	$1.3576 \cdot 10^{-9}$	$4.3857 \cdot 10^{-11}$
0.02	$8.9142 \cdot 10^{-9}$	$8.9111 \cdot 10^{-9}$	$3.5634 \cdot 10^{-11}$	$2.8583 \cdot 10^{-12}$
0.01	$1.1232 \cdot 10^{-9}$	$1.1228 \cdot 10^{-9}$	$2.2457 \cdot 10^{-12}$	$3.5945 \cdot 10^{-13}$

Table 11. Global error vs. step size for (4.6).

h	K3	BS3	RK4	QT3
0.1	0.222536	0.257829	0.267671	0.400077
0.05	0.353233	0.458556	0.455446	0.771582
0.02	0.86932	1.096014	1.093489	1.805714
0.01	1.571522	2.134188	2.135892	3.661295

Table 12. Run time (in seconds) vs. step size for (4.6).

An equation involving a sine function. The following initial value problem is qualitatively similar to the logistic equation as well. Consider

$$\begin{cases} \dot{y} = \sin(y), \\ y|_{t=0} = 0.01, \end{cases} \qquad 0 \le t \le 1. \tag{4.6}$$

The exact solution is

$$y(t) = 2\arctan(\tan(0.005)e^t), \quad 0 \le t \le 1.$$

The results are displayed in Tables 11 and 12, and in Figures 12 and 13.

Conclusion. In the tested cases, the global error of our third-order QT3 method is comparable to and often smaller by several orders of magnitude than the global error of the other tested methods of the same order from the Runge–Kutta family. We even observed it to be smaller or comparable to the global error of the classical

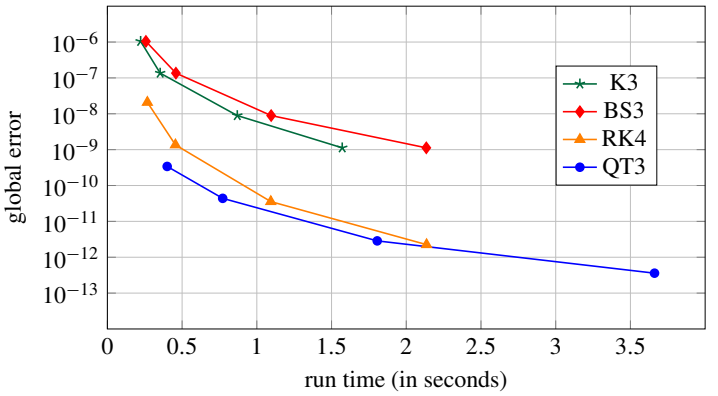

Figure 12. Work-precision diagram for (4.6) from Tables 11 and 12.

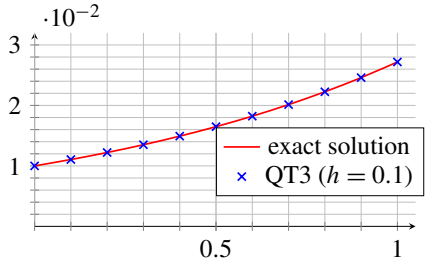

Figure 13. Exact solution and QT3 values for (4.6).

Runge–Kutta method of order 4 in most cases. This effect is most pronounced near equilibrium solutions of the tested equations.

Due to the complexity of the formulas the computational cost of QT3 is high. QT3 emphasizes accuracy over computational cost. Nonetheless, as exhibited by the work-precision data for the tested equations, QT3 not only exhibits superior accuracy but also efficiency for approximating solutions to ODEs with initial values near equilibrium solutions as larger step sizes and fewer computations suffice. QT3 thus may be particularly useful in stiff situations (we note that (4.5) is discussed in [Moler 2003] in this context).

In nonstiff situations, such as the initial value problem (4.3) for the Bernoulli equation, the increased computational cost of QT3 over standard algorithms such as BS3 does not appear to be adequately compensated for by gains in accuracy.

Appendix A: Convergence of one-step methods

Let $D \subset \mathbb{R}$ be open, and suppose $f : D \to \mathbb{R}$ satisfies a local Lipschitz condition in D. Let $y : [0, T] \to D$ be the solution to the initial value problem

$$\begin{cases} \dot{y}(t) = f(y(t)) & \text{on } 0 \le t \le T, \\ y|_{t=0} = y_0 \in D. \end{cases}$$

Theorem A.1 below is a general convergence result of abstract numerical one-step methods for the approximation of the solution y on partitions of the interval $[0, T]$; see, for example, [Kress 1998, Section 10.3]. It is the basis for proving Theorem 2.9 in Section 2. We restrict our attention to equidistant partitions of step size $h > 0$.

Theorem A.1. *Let $K \Subset D$ be a compact neighborhood with $y([0, T]) \subset \mathring{K}$, and let*

$$\Phi : [0, h_0] \times K \to \mathbb{R}$$

be continuous, $h_0 > 0$. Assume:

• ***Consistency***: *$\Phi(0, y) = y$ for all $y \in K$, and $\partial\Phi/\partial h : (0, h_0) \times K \to \mathbb{R}$ exists and extends to a continuous function on $[0, h_0] \times K$ such that $(\partial\Phi/\partial h)(0, y) = f(y)$ for all $y \in K$.*

• ***Lipschitz condition***: *the function $\partial\Phi/\partial h : [0, h_0] \times K \to \mathbb{R}$ satisfies a Lipschitz condition with respect to y; i.e., there exists a constant $L > 0$ such that*

$$\left| \frac{\partial\Phi}{\partial h}(h, y) - \frac{\partial\Phi}{\partial h}(h, y') \right| \le L|y - y'|$$

for all $0 \le h \le h_0$ and $y, y' \in K$.

• ***Local truncation error***: *there exists $p \ge 1$ and a constant $C \ge 0$ independent of $0 \le h \le h_0$ and $0 \le t \le T$ such that*

$$|y(t + h) - \Phi(h, y(t))| \le Ch^{p+1}$$

whenever $0 \le t + h \le T$.

Then Φ yields a one-step method of order p for the approximation of y on $[0, T]$; i.e., there exist $N_0 \in \mathbb{N}$ and a constant $M \ge 0$ such that for all $N \ge N_0$, $h = T/N$ the following holds:

The sequence of numbers $y_0^{(N)}, \dots, y_N^{(N)}$ defined via

$$\begin{cases} y_0^{(N)} = y_0, \\ y_{n+1}^{(N)} = \Phi(h, y_n^{(N)}), \quad n = 0, \dots, N - 1, \end{cases}$$

is well-defined, all $y_n^{(N)} \in \mathring{K}$, and the global error satisfies

$$\max_{n=0}^{N} |y(nh) - y_n^{(N)}| \le Mh^p. \tag{A.2}$$

A valid choice for the constant in (A.2) *is $M = (C/L)(e^{LT} - 1)$.*

Appendix B: Differential equations depending on parameters

Let $\Lambda \subset \mathbb{R}^q$ be open, and $V \subset \mathbb{R}$ be an open interval with $0 \in V$. Suppose $F(w; \lambda)$ is continuously differentiable with respect to the variables $(w; \lambda) \in V \times \Lambda$. Consider

the family of ordinary differential equations

$$\begin{cases} \dfrac{\partial w}{\partial t}(t;\lambda) = F(w(t;\lambda);\lambda), \\ w(0;\lambda) = 0, \end{cases} \tag{B.1}$$

for the unknown function $t \mapsto w(t;\lambda)$ depending on the parameter $\lambda \in \Lambda$. The following holds; see [Walter 1998].

Theorem B.2. *For each $\lambda \in \Lambda$ there exists a unique maximally extended solution*

$$w(\,\cdot\,;\lambda) : (t_{\min}(\lambda), t_{\max}(\lambda)) \to V$$

to (B.1), *where $-\infty \le t_{\min}(\lambda) < 0 < t_{\max}(\lambda) \le \infty$.*

The functions $t_{\max}, t_{\min} : \Lambda \to \mathbb{R} \cup \{\pm\infty\}$ are lower and upper semicontinuous, respectively, and the set

$$U_{\max} = \{(t,\lambda) : \lambda \in \Lambda, t_{\min}(\lambda) < t < t_{\max}(\lambda)\} \subset \mathbb{R} \times \mathbb{R}^q$$

is open. The solution w to (B.1) *defines a map $U_{\max} \to V$, and both w and $\partial w/\partial t$ are continuously differentiable in $U_{\max}$. The partial derivatives of w satisfy*

$$\begin{aligned} \frac{\partial w}{\partial t}(t;\lambda) &= F(w(t;\lambda);\lambda) \quad (\textit{this is just } (\text{B.1})), \\ (\nabla_\lambda w)(t;\lambda) &= \int_0^t e^{\int_s^t F_w(w(u;\lambda);\lambda)\,du} (\nabla_\lambda F)(w(s;\lambda);\lambda)\,ds. \end{aligned} \tag{B.3}$$

In particular, if F is more than once continuously differentiable, then so is w, and formulas for higher partial derivatives of w follow from (B.3) *with the chain rule.*

Remark B.4. The upper and lower semicontinuity of the endpoint functions of the maximal existence interval follow from the openness of $U_{\max}$. Semicontinuity implies that $t_{\min}$ attains its maximum value $t_{\min}(K) \in \mathbb{R} \cup \{-\infty\}$ and $t_{\max}$ attains its minimum value $t_{\max}(K) \in \mathbb{R} \cup \{\infty\}$ on every compact subset $K \Subset \Lambda$. In particular, $w(t;\lambda)$ is defined (and differentiable) for all $(t;\lambda) \in (t_{\min}(K), t_{\max}(K)) \times K$. Thus, for every compact subset $K \Subset \Lambda$, we are guaranteed that $w(t;\lambda)$ exists on $[0, T] \times K$ for some $T > 0$ (depending on K). We make use of this in the theoretical Section 2 of this paper.

References

[Bogacki and Shampine 1989] P. Bogacki and L. F. Shampine, "A 3(2) pair of Runge–Kutta formulas", *Appl. Math. Lett.* **2**:4 (1989), 321–325. MR Zbl

[Butcher 2008] J. C. Butcher, *Numerical methods for ordinary differential equations*, 2nd ed., John Wiley & Sons, Chichester, 2008. MR Zbl

[Corless et al. 1996] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth, "On the Lambert W function", *Adv. Comput. Math.* **5**:4 (1996), 329–359. MR Zbl

[Hairer et al. 1993] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving ordinary differential equations, I: Nonstiff problems*, 2nd ed., Springer Series in Computational Mathematics **8**, Springer, 1993. MR Zbl

[Hochbruck and Ostermann 2010] M. Hochbruck and A. Ostermann, "Exponential integrators", *Acta Numer.* **19** (2010), 209–286. MR Zbl

[Kassam and Trefethen 2005] A.-K. Kassam and L. N. Trefethen, "Fourth-order time-stepping for stiff PDEs", *SIAM J. Sci. Comput.* **26**:4 (2005), 1214–1233. MR Zbl

[Kress 1998] R. Kress, *Numerical analysis*, Graduate Texts in Mathematics **181**, Springer, 1998. MR Zbl

[Mickens 1994] R. E. Mickens, *Nonstandard finite difference models of differential equations*, World Scientific, River Edge, NJ, 1994. MR Zbl

[Mickens 2000] R. E. Mickens, "Nonstandard finite difference schemes", pp. 1–54 in *Applications of nonstandard finite difference schemes* (Atlanta, GA, 1999), edited by R. E. Mickens, World Scientific, River Edge, NJ, 2000. MR Zbl

[Moler 2003] C. Moler, "Stiff differential equations", blog post, 2003, https://www.mathworks.com/company/newsletters/articles/stiff-differential-equations.html.

[Patidar 2005] K. C. Patidar, "On the use of nonstandard finite difference methods", *J. Difference Equ. Appl.* **11**:8 (2005), 735–758. MR Zbl

[Ralston 1965] A. Ralston, *A first course in numerical analysis*, McGraw-Hill, New York, 1965. MR Zbl

[Shampine and Reichelt 1997] L. F. Shampine and M. W. Reichelt, "The MATLAB ODE suite", *SIAM J. Sci. Comput.* **18**:1 (1997), 1–22. MR Zbl

[Thieme 2003] H. R. Thieme, *Mathematics in population biology*, Princeton University Press, 2003. MR Zbl

[Vigo-Aguiar and Ramos 2011] J. Vigo-Aguiar and H. Ramos, "A numerical ODE solver that preserves the fixed points and their stability", *J. Comput. Appl. Math.* **235**:7 (2011), 1856–1867. MR Zbl

[Walter 1998] W. Walter, *Ordinary differential equations*, Graduate Texts in Mathematics **182**, Springer, 1998. MR Zbl

Received: 2019-03-19 Accepted: 2020-02-25

krainer@psu.edu *Penn State Altoona, Altoona, PA, United States*

cjz5145@psu.edu *Penn State Altoona, Altoona, PA, United States*

involve

msp.org/involve

INVOLVE YOUR STUDENTS IN RESEARCH

Involve showcases and encourages high-quality mathematical research involving students from all academic levels. The editorial board consists of mathematical scientists committed to nurturing student participation in research. Bridging the gap between the extremes of purely undergraduate research journals and mainstream research journals, *Involve* provides a venue to mathematicians wishing to encourage the creative involvement of students.

Cover: Alex Scorpan

See inside back cover or msp.org/involve for submission instructions. The subscription price for 2020 is US $205/year for the electronic version, and $275/year (+$35, if shipping outside the US) for print and electronic. Subscriptions, requests for back issues and changes of subscriber address should be sent to MSP.

Involve (ISSN 1944-4184 electronic, 1944-4176 printed) at Mathematical Sciences Publishers, 798 Evans Hall #3840, c/o University of California, Berkeley, CA 94720-3840, is published continuously online. Periodical rate postage paid at Berkeley, CA 94704, and additional mailing offices.

Involve peer review and production are managed by EditFlow® from Mathematical Sciences Publishers.

PUBLISHED BY

mathematical sciences publishers

nonprofit scientific publishing

http://msp.org/

involve

2020 vol. 13 no. 2

1944-4176(2020)13:2;1-S