Time-transformations for the event location in discontinuous ODEs

L. Lopez
Dipartimento di Matematica
Università di Bari
luciano.lopez@uniba.it

S. Maset
Dipartimento di Matematica e Geoscienze
Università di Trieste
maset@units.it

June 15, 2017

Abstract

In this paper, we consider numerical methods for the location of events of ordinary differential equations. These methods are based on particular changes of the independent variable, called time-transformations. Such a time-transformation reduces the integration of an equation up to the unknown point, where an event occurs, to the integration of another equation up to a known point. This known point corresponds to the unknown point by means of the time-transformation. This approach extends the one proposed in [12] but our generalization permits, amongst other things, to deal with situations where the solution approaches to the event in a tangential way. Moreover, we also propose to use this approach in a different manner with respect to [12].

1 Introduction

Recently the topic of discontinuous ordinary differential equations (ODEs) has attracted a lot of interest either from a theoretical or computational point of view also because of its different applications (see for example [1, 8, 10, 11, 15, 16, 17]).

An important task, in the numerical solution of discontinuous ODEs is the location of the events on the discontinuity surface (see for instance [3, 6, 7, 9, 12, 13, 18]). Here we propose a time-transformation method to compute efficiently such event points.

Let us consider the region

$$R := \{x \in \mathbb{R}^n : h(x) < 0\},$$

with border

$$\Sigma := \partial R = \{x \in \mathbb{R}^n : h(x) = 0\},$$

where $h : \mathbb{R}^n \to \mathbb{R}$, and the ordinary differential equation

$$\begin{cases} x'(t) = f(x(t)), \ t \geq 0, \\ x(0) = x_0, \end{cases} \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$ and $x_0 \in \mathbb{R}^n$ is such that $x_0 \in R$. We assume that h and f are sufficiently smooth functions and that the ODE (1) has a unique solution x. We observe that, in the applications, $h(x)$ is often linear or quadratic with respect to x (see [10]).

The ODE (1) has to be integrated up to the first point $t_f > 0$, which is unknown, such that

$$x(t_f) \in \Sigma, \tag{2}$$

i.e. we have to locate the event (2) during the integration of the ODE (1).

Assume that a numerical integration of (1) is accomplished over a mesh

$$t_0 < t_1 < t_2 < \cdots \tag{3}$$

with stepsizes $\tau_{n+1} = t_{n+1} - t_n$, $n = 0, 1, 2, \ldots$, by a Runge-Kutta (RK) method (A, b, c). The method yields a sequence $\{x_n\}$, $n = 0, 1, 2, \ldots$, where x_n is an approximation of $x(t_n)$, recursively given by

$$\begin{aligned} x_{n+1} &= x_n + \tau_{n+1} \sum_{i=1}^{\nu} b_i f\left(X_i^{n+1}\right) \\ X_i^{n+1} &= x_n + \tau_{n+1} \sum_{j=1}^{\nu} a_{ij} f\left(X_j^{n+1}\right), \ i = 1, \ldots, \nu. \end{aligned}$$

The classical approach for locating the event (2), described for example in [6] and [18], is as follows. We proceed up to the first point $t_{\overline{n}+1}$ such that

$$h(x_{\overline{n}})\, h(x_{\overline{n}+1}) < 0.$$

Then, by using a continuous numerical solution $\eta(t)$, $t \in [t_{\overline{n}}, t_{\overline{n}+1}]$, given for example by a Continuous RK method $(A, b(\cdot), c)$ as

$$\eta(x_{\overline{n}} + \theta \tau_{\overline{n}+1}) = x_{\overline{n}} + \tau_{\overline{n}+1} \sum_{i=1}^{\nu} b_i(\theta) f\left(X_i^{\overline{n}+1}\right), \ \theta \in [0, 1], \tag{4}$$

an approximation $\widetilde{t}_f$ of t_f can be obtained by solving the scalar equation

$$h(\eta(t)) = 0. \tag{5}$$

Of course, by assuming that the equation (5) is solved exactly, the order of the approximations $\widetilde{t}_f$ of t_f and $\eta\left(\widetilde{t}_f\right)$ of $x(t_f)$ is the order of the continuous approximation η of x, which is in general less than the order p of the RK method, the order with which are computed the discrete approximations x_n.

In order to recover the order p, one can consider as approximation of t_f a new unknown mesh point $\widehat{t}_{\overline{n}+1}$ (successive to $t_{\overline{n}}$) and as an approximation of $x\left(t_f\right)$ a corresponding new discrete approximation $\widehat{x}_{\overline{n}+1}$, at the mesh point $\widehat{t}_{\overline{n}+1}$, such that $h\left(\widehat{x}_{\overline{n}+1}\right)=0$. So, we have

$$\begin{aligned} \widehat{t}_{\overline{n}+1} &= t_{\overline{n}}+\widehat{\tau}_{\overline{n}+1} \\ \widehat{x}_{\overline{n}+1} &= x_{\overline{n}}+\widehat{\tau}_{\overline{n}+1}\sum_{i=1}^{\nu} b_i \widehat{K}_i^{\overline{n}+1}, \end{aligned}$$

where $\widehat{K}_i^{\overline{n}+1}$, $i=1,\ldots,\nu$, and $\widehat{\tau}_{\overline{n}+1}$ are obtained by solving the equations

$$\begin{aligned} &\widehat{K}_i^{\overline{n}+1} = f\left(x_{\overline{n}}+\widehat{\tau}_{\overline{n}+1}\sum_{i=1}^{\nu} a_{ij}\widehat{K}_j^{\overline{n}+1}\right),\ i=1,\ldots,\nu, \\ &h\left(x_{\overline{n}}+\widehat{\tau}_{\overline{n}+1}\sum_{i=1}^{\nu} b_i\widehat{K}_i^{\overline{n}+1}\right)=0. \end{aligned} \tag{6}$$

This approach is described in [2, 14] in the context of the computation of breaking points of delay differential equations. Note that now we have to solve a square system of $n\nu+1$ scalar equations instead of the sole scalar equation (5), even if we are using an explicit method. So, this approach is particularly suitable when an implicit method is used, as in case of stiff problems.

Recently, a new approach to the event location has been introduced in [12] where, by a suitable change of the variable time t, the ODE (1) is reduced to another ODE and the location of the event (2) is known in advance.

In the present paper, we propose a generalization of this approach which permits, amongst other things, to deal with situations where the solution x lands on the border Σ in a tangential way. Moreover, we also propose to use this approach in a different manner with respect to [12].

Here is the plan of the paper: Section 2 describes the generalized approach and the new proposed manner to use it; Section 3 contains a convergence analysis for the new manner; Section 4 study exact numerical landing on the border Σ and one-sided integration; finally, Section 5 deals with tangential landing on Σ and conclusions are drawn in Section 6.

2 Time-transformations

We apply to the event location problem of the ODE (1) the idea of the time-transformations introduced in [4, 5] in the context of the delay differential equations. The resulting approach includes as a particular case the approach presented in [12].

The idea is to introduce a strictly increasing function $\alpha:[s_0,0]\to[0,t_f]$, where $s_0<0$, such that $\alpha(s_0)=0$ and $\alpha(0)=t_f$, and then to set

$$y\left(s\right):=x\left(\alpha\left(s\right)\right),\ s\in\left[s_0,0\right],$$

where x is the solution of (1). Then, y satisfies

$$y'(s) = f(y(s))\,\alpha'(s)\,,\ s \in [s_0, 0]\,.$$

The function α is called a *time-transformation*.

Now, we look for a time-transformation α such that

$$h(x(\alpha(s))) = h(y(s)) = \kappa(s)\,,\ s \in [s_0, 0]\,, \tag{7}$$

where $\kappa : [s_0, 0] \to [h(x_0), 0]$ is a given strictly increasing function of class C^1 such that $\kappa(s_0) = h(x_0) < 0$ and $\kappa(0) = 0$.

In the following, we assume there exist $\delta, c_0 > 0$ such that

$$h'(x)\,f(x) \geq \delta,\ x \in R \cup \Sigma \text{ such that } h(x) > -c_0, \tag{8}$$

where $h'(x)$ is the row-vector gradient of h at the point x. Moreover, we assume $h(x_0) > -c_0$.

By differentiating (7) we obtain

$$h'(y(s))\,y'(s) = h'(y(s))\,f(y(s))\,\alpha'(s) = \kappa'(s)$$

and then the transformed ODE

$$\begin{cases} \begin{bmatrix} y'(s) \\ \alpha'(s) \end{bmatrix} = \frac{\kappa'(s)}{h'(y(s))f(y(s))} \begin{bmatrix} f(y(s)) \\ 1 \end{bmatrix},\ s \in [s_0, 0]\,, \\ \\ \begin{bmatrix} y(s_0) \\ \alpha(s_0) \end{bmatrix} = \begin{bmatrix} x_0 \\ 0 \end{bmatrix} \end{cases} \tag{9}$$

where

$$s_0 = \kappa^{-1}(h(x_0)).$$

Observe that in (9) we have

$$h'(y(s))\,f(y(s)) \geq \delta,\ s \in [s_0, 0]\,,$$

since (8) holds and

$$h(y(s)) = \kappa(s) \in [h(x_0), 0] \subseteq (-c_0, 0],\ s \in [s_0, 0]\,.$$

By integrating the ODE (9), we obtain y and α and then x can be reconstructed by

$$y(s) = x(\alpha(s))\,,\ s \in [s_0, 0]\,.$$

We have

$$h(y(s)) = \kappa(s),\ s \in [s_0, 0]\,,$$

and this means that in the transformed ODE (9) the solution y approaches to the border Σ, where it lands at $s = 0$, in the manner prescribed in advance by the function κ.

In the s-time, the event is located at 0 with value $y(0)$. In the original t-time, the event is located at $t_f = \alpha(0)$ with value $x(t_f) = y(0)$.

The approach presented in [12] corresponds to set

$$\kappa(s) = s,\ s \in [s_0, 0], \tag{10}$$

where $s_0 = h(x_0)$.

By numerically integrating the ODE (9) by a RK (A, b, c) method over the mesh

$$s_0 = \kappa^{-1}(h(x_0)) < s_1 < \cdots < s_N = 0,$$

with stepsizes $\sigma_{n+1} = s_{n+1} - s_n$, $n = 0, 1, \ldots, N-1$, we obtain the scheme

$$\begin{aligned} &\begin{bmatrix} y_{n+1} \\ \alpha_{n+1} \end{bmatrix} = \begin{bmatrix} y_n \\ \alpha_n \end{bmatrix} + \sigma_{n+1} \sum_{i=1}^{\nu} b_i \frac{\kappa'\left(s_i^{n+1}\right)}{h'\left(Y_i^{n+1}\right) f\left(Y_i^{n+1}\right)} \begin{bmatrix} f\left(Y_i^{n+1}\right) \\ 1 \end{bmatrix} \\ &\begin{bmatrix} Y_i^{n+1} \\ \Lambda_i^{n+1} \end{bmatrix} = \begin{bmatrix} y_n \\ \alpha_n \end{bmatrix} + \sigma_{n+1} \sum_{j=1}^{\nu} a_{ij} \frac{\kappa'\left(s_j^{n+1}\right)}{h'\left(Y_j^{n+1}\right) f\left(Y_j^{n+1}\right)} \begin{bmatrix} f\left(Y_j^{n+1}\right) \\ 1 \end{bmatrix} \\ &i = 1, \ldots, \nu \ \text{ and } \ n = 0, 1, \ldots, N-1 \end{aligned}$$

where

$$s_i^{n+1} := s_n + c_i \sigma_{n+1},\ i = 1, \ldots, \nu.$$

Numerically, in the t-time the event is located at $\alpha_N \approx \alpha(0) = t_f$ with value $y_N \approx y(0) = x(t_f)$.

Observe that in this approach no additional algebraic equation like (5) or (6) has to be solved in order to locate the event (2): if an explict RK method is used for integrating (9), then the process of localizations turns out to be explicit.

On the other hand, we can observe that dimension of the state space is augmented by one, since now the time-transformation appears as component in the new state space. However, we can avoid to compute the time transformation if we are only interested in state $y(t_f)$ when the event happens and not to the time t_f.

In the following, as concrete examples of function κ, we consider the functions $\kappa_{m,C}$ given by

$$\kappa_{m,C}(s) = -C(-s)^m = -C|s|^m,\ s \in [s_0, 0], \tag{11}$$

where $m \geq 1$ and $C > 0$. They are the simplest examples of function κ that one can conceive: they are strictly increasing one-term polynomial functions with value 0 at 0 and with negative values at negative arguments.

2.1 The procedures A and B

We can use this approach of the time-transformations by following two procedures A and B now described.

A. Transform the problem from the beginning, as described up to now, by solving (9).

B. Numerically integrate the original equation (1), not the transformed equation (9), as previously described in Section 1 up to the first point $t_{\overline{n}+1}$ such that
$$h\left(x_{\overline{n}}\right) h\left(x_{\overline{n}+1}\right) < 0.$$
Only now, the problem is transformed by solving
$$\left\{ \begin{array}{l} \left[\begin{array}{c} y'(s) \\ \alpha'(s) \end{array} \right] = \frac{\kappa'(s)}{h'(y(s))f(y(s))} \left[\begin{array}{c} f(y(s)) \\ 1 \end{array} \right], \ s \in [s_0, 0], \\ \\ \left[\begin{array}{c} y(s_0) \\ \alpha(s_0) \end{array} \right] = \left[\begin{array}{c} x_{\overline{n}} \\ 0 \end{array} \right] \end{array} \right.$$
where
$$s_0 = \kappa^{-1}(h(x_{\overline{n}})).$$
One step of a RK method now provides approximations α_1 of t_f and y_1 of $x(t_f)$.

The paper [12] deals with the particular time-transformation (10) as applied in the procedure A. The present paper deals with a general time-transformation as applied in both procedures A and B.

In the procedure B, the numerical integration over one step by a RK method (A, b, c) is given by
$$\begin{array}{l} \left[\begin{array}{c} y_1 \\ \alpha_1 \end{array} \right] = \left[\begin{array}{c} x_{\overline{n}} \\ 0 \end{array} \right] + (-s_0) \sum_{i=1}^{\nu} b_i \frac{\kappa'\left(s_i^1\right)}{h'\left(Y_i^1\right) f\left(Y_i^1\right)} \left[\begin{array}{c} f\left(Y_i^1\right) \\ 1 \end{array} \right] \\ \left[\begin{array}{c} Y_i^1 \\ \Lambda_i^1 \end{array} \right] = \left[\begin{array}{c} x_{\overline{n}} \\ 0 \end{array} \right] + (-s_0) \sum_{j=1}^{\nu} a_{ij} \frac{\kappa'\left(s_j^1\right)}{h'\left(Y_j^1\right) f\left(Y_j^1\right)} \left[\begin{array}{c} f\left(Y_j^1\right) \\ 1 \end{array} \right] \\ i = 1, \ldots, \nu \end{array} \tag{12}$$
where $s_0 = \kappa^{-1}(h(x_{\overline{n}}))$ and
$$s_i^1 := s_0 + c_i(-s_0) = (1 - c_i)s_0, \ i = 1, \ldots, \nu.$$

Next theorem shows that, for functions $\kappa_{m,C}$ in (11), we always can reduce the situation to one with $\kappa(s) = s$ in the procedure B.

Theorem 1 *Assume that a RK method (A, b, c) is applied over one step to the problem*
$$\left\{ \begin{array}{l} z'(s) = \kappa'(s)G(z(s)), \ s \in [s_0, 0], \\ z(s_0) = z_0, \end{array} \right. \tag{13}$$

with $\kappa = \kappa_{m,C}$. The numerical solution z_1 and the stage values Z_i^1, $i = 1, \ldots, \nu$, are the numerical solution and the stage values, respectively, provided by the RK method $\left(A^{(m)}, b^{(m)}, c\right)$ with

$$\begin{aligned} b_i^{(m)} &= b_i m \left(1 - c_i\right)^{m-1}, \ i = 1, \ldots, \nu, \\ a_{ij}^{(m)} &= a_{ij} m \left(1 - c_j\right)^{m-1}, \ i, j = 1, \ldots, \nu, \end{aligned}$$

as applied over one step to

$$\begin{cases} u'(s) = G(u(s)), \ s \in [\kappa_{m,C}(s_0), 0], \\ u(\kappa_{m,C}(s_0)) = z_0 \end{cases} \tag{14}$$

which is a problem (13) with $\kappa(s) = s$.

Proof. By applying the RK method (A, b, c) over one step to the problem (13) with $\kappa = \kappa_{m,C}$, we obtain

$$\begin{aligned} z_1 &= z_0 + (-s_0) \sum_{i=1}^{\nu} b_i C m \left(-s_i^1\right)^{m-1} G\left(Z_i^1\right) \\ &= z_0 + (-s_0) \sum_{i=1}^{\nu} b_i C m \left(\left(1 - c_i\right)(-s_0)\right)^{m-1} G\left(Z_i^1\right) \\ &= z_0 + \underbrace{C(-s_0)^m}_{=-\kappa_{m,C}(s_0)} \sum_{i=1}^{\nu} b_i m \left(1 - c_i\right)^{m-1} G\left(Z_i^1\right) \end{aligned}$$

and, analougsly,

$$Z_i^1 = z_0 + \underbrace{C(-s_0)^m}_{=-\kappa_{m,C}(s_0)} \sum_{j=1}^{\nu} a_{ij} m \left(1 - c_j\right)^{m-1} G\left(Z_j^1\right), \ i = 1, \ldots, \nu.$$

This means that z_1 and Z_i^1, $i = 1, \ldots, \nu$, are the numerical solution and the stage values, respectively, provided by the RK method $\left(A^{(m)}, b^{(m)}, c\right)$ when it is applied over one step to the problem (14). ■

As a consequence of this result, we obtain that, when in the procedure B we solve over one step the transformed equation with

$$\kappa = \kappa_{m,C} \quad \text{and} \quad s_0 = \kappa_{m,C}^{-1}(h(x_{\overline{n}}))$$

by the RK method (A, b, c), the numerical solution and the stage values are the same how to numerically solve over one step the transformed equation with

$$\kappa(s) = s \quad \text{and} \quad s_0 = \kappa_{m,C}(\kappa_{m,C}^{-1}(h(x_{\overline{n}}))) = h(x_{\overline{n}})$$

by the RK method $\left(A^{(m)}, b^{(m)}, c\right)$.

Remark 2 *This means that in the procedure* B *we cannot obtain any advantage by choosing a function $\kappa = \kappa_{m,C}$ different from $\kappa(s) = s$, since the change from $k(s) = s$ to $\kappa = \kappa_{m,C}$ corresponds to use $k(s) = s$ with another RK method. For this reason, from now on we always use $k(s) = s$ in the procedure* B.

On the other hand, in the procedure A we can have advantages in using a function κ different from $\kappa(s) = s$. For example, in the situation of tangential landing on the border Σ, as described in Section 5. Another situation, where a function κ different from $k(s) = s$ can be used, is when we are interested not only in the event (2) but also in a sequence of events

$$h(x(t)) = h_i,\ i = 1, 2, \ldots, q, \tag{15}$$

where

$$h(x_0) < h_1 < h_2 < \cdots < h_q < 0,$$

because, for example, we need to check the approach to the border Σ. In this case, we can integrate the transformed equation (9) with a function κ such that

$$\kappa\left(\frac{q+1-i}{q+1}s_0\right) = \kappa\left(s_0 + \frac{i}{q+1}(-s_0)\right) = h_i,\ i = 1, 2, \ldots, q.$$

So, by numerically integrating the transformed equation with constant stepsize

$$\sigma = \frac{-s_0}{(q+1)M},$$

where M is a positive integer, the events (15) will be located at the s-times s_{iM}, $i = 1, 2, \ldots, q$, with numerical value y_{iM}.

3 Convergence results

As for the procedure A, under the assumption that κ is sufficientely smooth on $[s_0, 0]$, we have

$$\left\|\begin{bmatrix} y_N \\ \alpha_N \end{bmatrix} - \begin{bmatrix} x(t_f) \\ t_f \end{bmatrix}\right\|_\infty = O\left(\sigma^q\right),\ \sigma \to 0,$$

where σ is the maximum of the stepsizes σ_{n+1}, $n = 0, 1, \ldots, N-1$, and q is the order of the RK method used in the integration of the transformed equation.

As for the procedure B, we begin with the following lemma. In this lemma and in the following, τ denotes the maximum stepsize in the mesh (3) up to $t_{\overline{n}+1}$.

Lemma 3 *In the procedure* B *we have*

$$|h\left(x_{\overline{n}}\right)| = O\left(\tau_{\overline{n}+1}\right),\ \tau \to 0.$$

Proof. Let η be the continuous approximation given in (4) and let $\widetilde{t}_f$ such that

$$h\left(\eta\left(\widetilde{t}_f\right)\right) = 0.$$

Then

$$\left|h\left(x_{\overline{n}}\right)\right| = \left|h\left(x_{\overline{n}}\right) - h\left(\eta\left(\widetilde{t}_f\right)\right)\right| \leq L\left|x_{\overline{n}} - \eta\left(\widetilde{t}_f\right)\right|,$$

where L is a Lipschitz constant of the function h in a suitable neighboorhood of $x\left(t_f\right)$, and, by recalling (4),

$$\left|x_{\overline{n}} - \eta\left(\widetilde{t}_f\right)\right| \leq \tau_{\overline{n}+1} \sum_{i=1}^{\nu} \left|b_i\left(\theta\right)\right| \left|f\left(Y_i^{\overline{n}+1}\right)\right| = O\left(\tau_{\overline{n}+1}\right), \ \tau \to 0,$$

where $\widetilde{t}_f = t_{\overline{n}} + \theta\tau_{\overline{n}+1}$. ■

Here is the convergence result for the procedure B.

Theorem 4 *In the procedure* B *with* $\kappa(s) = s$ *assume that:*

B1) the integration of the original equation is accomplished by a RK method of order p *over the mesh (3);*

B2) the integration of the transformed equation over one step is accomplished by a RK method of local order $q+1$.

Then

$$\left\| \begin{bmatrix} y_1 \\ \alpha_1 \end{bmatrix} - \begin{bmatrix} x(t_f) \\ t_f \end{bmatrix} \right\|_\infty = O\left(\tau^{\min\{p,q+1\}}\right), \ \tau \to 0.$$

Proof. By the previous lemma, we obtain

$$\left|s_0\right| = O\left(\left|h(x_{\overline{n}})\right|\right) = O\left(\tau_{\overline{n}+1}\right), \ \tau \to 0.$$

Now, let x^* be the solution of

$$\begin{cases} \left(x^*\right)'(t) = f\left(x^*\left(t\right)\right), \ t \geq t_{\overline{n}}, \\ x^*\left(t_{\overline{n}}\right) = x_{\overline{n}} \end{cases}$$

and let t_f^* be the first point such that

$$x^*\left(t_f^*\right) \in \Sigma.$$

By B2) we have

$$\left\| \begin{bmatrix} y_1 \\ \alpha_1 \end{bmatrix} - \begin{bmatrix} x^*\left(t_f^*\right) \\ t_f^* \end{bmatrix} \right\|_\infty = O\left(\left|s_0\right|^{q+1}\right) = O\left(\tau_{\overline{n}+1}^{q+1}\right), \ \tau \to 0.$$

On the other hand, by B1) we have

$$\left\| \begin{bmatrix} x^*\left(t_f^*\right) \\ t_f^* \end{bmatrix} - \begin{bmatrix} x\left(t_f\right) \\ t_f \end{bmatrix} \right\|_\infty = O\left(\left\|x_{\overline{n}} - x\left(t_n\right)\right\|_\infty\right) = O\left(\tau^p\right), \ \tau \to 0,$$

and so

$$\left\| \begin{bmatrix} y_1 \\ \alpha_1 \end{bmatrix} - \begin{bmatrix} x(t_f) \\ t_f \end{bmatrix} \right\|_\infty = O(\tau^p) + O\left(\tau_{\overline{n}+1}^{q+1}\right), \ \tau \to 0.$$

■

The previous proposition says that, in case of an explicit RK method integrating the original equation with order p and of an explicit RK method integrating the transformed equation over one step with order $q+1 \geq p$, we can explicitely find approximations of $x(t_f)$ and t_f of order p.

Example 5 *Consider the problem taken from [12]*

$$f(x) = \left(x_2, -x_1 + \frac{1}{1.2 - x_2} \right), \ t_0 = 0, \ x_0 = (-0.2, -0.2),$$
$$h(x) = x_1 + x_2 - 0.4 \ .$$

By integrating the original ODEs by the Heun method, whose order is $p = 2$, with constant stepsize $\tau = 10^{-2}$, we stop at

$$t_{\overline{n}} = 0.61, \ x_{\overline{n}} = (-0.12374, 0.51048).$$

By integrating the transformed equation for $\kappa(s) = s$ with the explicit Euler method, whose local order is $q + 1 = 2$, over one-step the event is numerically located at

$$t_f \approx s_1 = 0.61636, \ x(t_f) \approx y_1 = (-0.12049, 0.52049).$$

In Figure 1, we see the trajectory of the solution in the phase space. Observe that the landing on the border $h(x) = 0$ is not tangential. Indeed, we have $h(y_1)f(y_1) = 2.1126$.

Next table give the estimated errors

$$|\alpha_1 - t_f|, \| \ y_1 - x(t_f) \ \|_\infty$$

for $\tau = 10^{-k}$, $k = 1, \ldots, 4$, where t_f and $x(t_f)$ are estimated by using $\tau = 10^{-5}$.

τ	$\lvert\alpha_1 - t_f\rvert$	*ratios*	$\lVert y_1 - x(t_f)\rVert_\infty$	*ratios*
10^{-1}	$4.49 \cdot 10^{-4}$	13.4	$1.02 \cdot 10^{-3}$	50.0
10^{-2}	$3.35 \cdot 10^{-6}$	1449.1	$2.05 \cdot 10^{-5}$	152.7
10^{-3}	$2.31 \cdot 10^{-8}$	126.0	$1.33 \cdot 10^{-7}$	108.4
10^{-4}	$1.83 \cdot 10^{-10}$		$1.23 \cdot 10^{-9}$	

In this table and in the next tables, the i-th row of a column named "ratios" denotes the ratio between the errors in the previous column at the i-th and $i+1$-th rows. For a method of order p this ratio is expected to be about 10^p.

The whole method, given by the integration of the original equation by the Heun method and the integration of transformed equation by the explicit Euler method over one step, exhibits order $O\left(\tau^2\right)$ as predicted by the previous theorem (the geometric means of the ratios are 134.8 for $|\alpha_1 - t_f|$ and 93.9 for $\|y_1 - x(t_f)\|_\infty$).

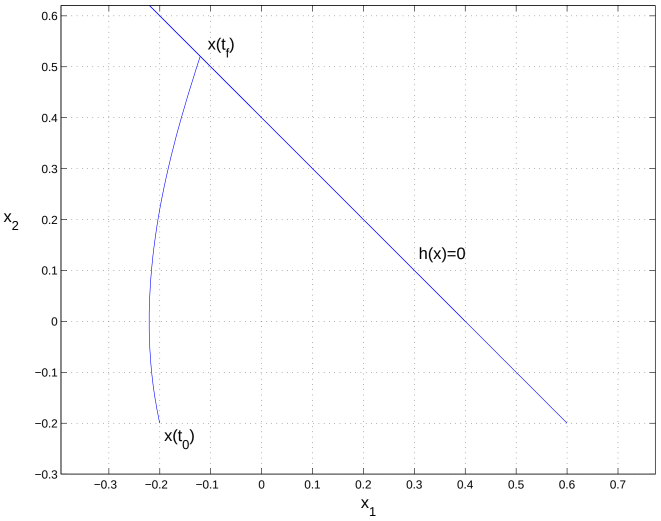

Figure 1: The trajectory $x(t)$ in the phase space.

4 Exact landing and one-sided integration

The fact $h\left(y_N\right)=0$ in the procedure A and $h(y_1)=0$ in the procedure B guarantee that the numerical integration of the transformed equation lands exactly on the border Σ. If a sliding motion takes place after the event, an exact landing on Σ is particularly important since the annoying phenomenon of numerical chattering can be avoided.

Moreover, in case of an explicit method, the facts

$$h(Y_i^{n+1})\leq 0,\ i=1,\ldots,\nu \text{ and } n=0,\ldots,N-1,$$

in the procedure A and

$$h(Y_i^{1})\leq 0,\ i=1,\ldots,\nu,$$

in the procedure B guarantee that the integration is *one-sided*, i.e. during the integration is never required to compute f on arguments outside $\Sigma\cup R$ in (12) (see [11]). A one-sided integration is particularly important when it is not easy to smoothly extend the function f outside $\Sigma\cup R$.

In this section, for a general RK method not necessarily explicit, we study these aspects in the cases of h linear, i.e.

$$h\left(x\right)=d^Tx+e,\ x\in\mathbb{R}^n,$$

where $d,e\in\mathbb{R}^n$, and h quadratic, i.e.

$$h\left(x\right)=x^TMx+d^Tx+e,\ x\in\mathbb{R}^n,$$

where $M \in \mathbb{R}^{n \times n}$ and $d, e \in \mathbb{R}^n$.

In the applications, often the surface Σ is defined by a linear or quadratic function h. This is the reason for which we consider, in detail, these two cases.

In the following the quadrature rule

$$(\beta - \alpha) \sum_{k=1}^{l} w_k f\left(\alpha + \gamma_k \left(\beta - \alpha\right)\right)$$

of weights w_k and nodes γ_k for the integral

$$\int_{\alpha}^{\beta} f\left(x\right) dx$$

is denoted by $\left(w_k, \gamma_k\right)_{k=1,\ldots,l}$.

4.1 The case h linear

Theorem 6 *Assume that h is linear and that the function κ in the transformed equation is a polynomial of degree m. Moreover, assume that a RK method (A, b, c) is used for the integration of the transformed equation.*

If the quadrature rule $\left(b_i, c_i\right)_{i=1,\ldots,\nu}$ has polynomial order $m-1$, then

$$h\left(y_{n+1}\right) = h\left(y_n\right) + \kappa\left(s_{n+1}\right) - \kappa\left(s_n\right),\ n = 0, 1, 2, \ldots$$

Moreover, for $i = 1, \ldots, \nu$ such that $c_i \neq 0$, if the quadrature rule $\left(\frac{a_{ij}}{c_i}, \frac{c_j}{c_i}\right)_{j=1,\ldots,\nu}$ has polynomial order $m-1$, then

$$h\left(Y_i^{n+1}\right) = h\left(y_n\right) + \kappa\left(s_i^{n+1}\right) - \kappa\left(s_n\right),\ n = 0, 1, 2, \ldots \tag{16}$$

Proof. Observe that

$$h'\left(x\right) = d^T,\ x \in \mathbb{R}^n,$$

and

$$h\left(x + z\right) = h\left(x\right) + d^T z,\ x, z \in \mathbb{R}^n.$$

We have

$$\begin{aligned} h\left(y_{n+1}\right) &= h\left(y_n + \sigma_{n+1} \sum_{i=1}^{\nu} b_i \frac{\kappa'\left(s_i^{n+1}\right) f\left(Y_i^{n+1}\right)}{d^T f\left(Y_i^{n+1}\right)}\right) \\ &= h\left(y_n\right) + \sigma_{n+1} \sum_{i=1}^{\nu} b_i \kappa'\left(s_i^{n+1}\right) \\ &= h\left(y_n\right) + \sigma_{n+1} \sum_{i=1}^{\nu} b_i \kappa'\left(s_n + c_i \sigma_{n+1}\right) \\ &= h\left(y_n\right) + \int_{s_n}^{s_{n+1}} \kappa'\left(s\right) ds \end{aligned}$$

if the quadrature rule $(b_i, c_i)_{i=1,\ldots,\nu}$ has polynomial order $m-1$. Moreover, for $i = 1, \ldots, \nu$ such that $c_i \neq 0$, we have

$$\begin{aligned} h\left(Y_i^{n+1}\right) &= h\left(y_n + \sigma_{n+1} \sum_{j=1}^{\nu} a_{ij} \frac{\kappa'\left(s_j^{n+1}\right) f\left(Y_j^{n+1}\right)}{d^T f\left(Y_j^{n+1}\right)}\right) \\ &= h\left(y_n\right) + \sigma_{n+1} \sum_{j=1}^{\nu} a_{ij} \kappa'\left(s_j^{n+1}\right) \\ &= h\left(y_n\right) + c_i \sigma_{n+1} \sum_{j=1}^{\nu} \frac{a_{ij}}{c_i} \kappa'\left(s_n + \frac{c_j}{c_i} c_i \sigma_{n+1}\right) \\ &= h\left(y_n\right) + \int_{s_n}^{s_i^{n+1}} \kappa'\left(s\right) ds \\ &= h\left(y_n\right) + \kappa\left(s_i^{n+1}\right) - \kappa\left(s_n\right) \end{aligned}$$

if the quadrature rule $\left(\frac{a_{ij}}{c_i}, \frac{c_j}{c_i}\right)_{i=1,\ldots,\nu}$ has polynomial order $m-1$. ■

Observe that (16) holds also for $c_i = 0$ if

$$Y_i^{n+1} = y_n,\ n = 0, 1, 2, \ldots$$

and this happens, for example, when $a_{ij} = 0$, $j = 1, \ldots, \nu$.

As a consequence of the previous theorem, in case of a linear function h, we can conclude as follows.

In the procedure A, if the function κ is a polynomial of degree m and the transformed equation is integrated by a RK method (A, b, c) such that the quadrature rule $(b_i, c_i)_{i=1,\ldots,\nu}$ has polynomial order $m-1$, then

$$h\left(y_n\right) = \kappa\left(s_n\right),\ n = 0, 1, \ldots, N,$$

and so $h\left(y_N\right) = 0$. Moreover, if the quadrature rule $\left(\frac{a_{ij}}{c_i}, \frac{c_j}{c_i}\right)_{j=1,\ldots,\nu}$ has polynomial order $m-1$ for any $i = 1, \ldots, \nu$ such that $c_i \neq 0$ and

$$Y_i^{n+1} = y_n,\ n = 0, 1, \ldots, N-1,$$

for any $i = 1, \ldots, \nu$ such that $c_i = 0$, then

$$h(Y_i^{n+1}) = \kappa\left(s_i^{n+1}\right) \leq 0,\ i = 1, \ldots, \nu \text{ and } n = 0, \ldots, N-1. \qquad (17)$$

Observe that for an explicit RK method (A, b, c), the quadrature rule

$$\left(\frac{a_{2j}}{c_2}, \frac{c_j}{c_2}\right)_{j=1,\ldots,\nu}$$

relevant to the index $i = 2$ is the one-node quadrature rule of weight $\frac{a_{21}}{c_2}$ and node 0, whose polynomial order is 0. So, for an explicit RK method integrating the transformed equation in the procedure A, one cannot guarantee (17) in case of polynomial function κ of degree $m > 1$.

In the procedure B with $\kappa(s) = s$, if the transformed equation is integrated over one step by a RK method (A, b, c) such that

$$\sum_{i=1}^{\nu} b_i = 1$$

(so the quadrature rule $(b_i, c_i)_{i=1,\ldots,\nu}$ has polynomial order 0) and

$$\sum_{i=1}^{\nu} a_{ij} = c_i,\ i = 1, \ldots, \nu,$$

(so the quadrature rule $\left(\frac{a_{ij}}{c_i}, \frac{c_j}{c_i}\right)_{j=1,\ldots,\nu}$ has polynomial order 0 for $c_i \neq 0$ and $Y_i^{n+1} = y_n$ for $c_i = 0$), then

$$h\left(y_1\right) = 0$$

and

$$h(Y_i^1) = \kappa\left(s_i^1\right) \leq 0,\ i = 1, \ldots, \nu.$$

So, exact landing on Σ and one-sided integration can be obtained by explicit methods in the case of h linear for both procedures A and B.

4.2 The case h quadratic

Theorem 7 *Assume that h is quadratic and that the function κ in the transformed equation is a polynomial of degree m. Moreover, assume that a RK method (A, b, c) is used for the integration of the trasformed equation.*

If the quadrature rule $(b_i, c_i)_{i=1,\ldots,\nu}$ has polynomial order $m - 1$, then

$$\begin{aligned}
&h\left(y_{n+1}\right) = h\left(y_n\right) + \kappa\left(s_{n+1}\right) - \kappa\left(s_n\right) \\
&+\sigma_{n+1}^2 \sum_{i=1}^{\nu}\sum_{j=1}^{\nu}\left(b_i b_j - b_i a_{ij} - b_j a_{ji}\right)\frac{\kappa'\left(s_i^{n+1}\right)\kappa'\left(s_j^{n+1}\right) f\left(Y_i^{n+1}\right)^T M f\left(Y_j^{n+1}\right)}{h'\left(Y_i^{n+1}\right) f\left(Y_i^{n+1}\right)\cdot h'\left(Y_j^{n+1}\right) f\left(Y_j^{n+1}\right)} \\
&n = 0, 1, 2, \ldots
\end{aligned}$$

Proof. Observe that

$$\begin{aligned}
h\left(x+z\right) &= \left(x+z\right)^T M\left(x+z\right) + d^T\left(x+z\right) + e \\
&= x^T M x + d^T x + e + x^T M z + \underbrace{z^T M x}_{=x^T M^T z} + d^T z + z^T M z \\
&= h\left(x\right) + x^T\left(M + M^T\right) z + d^T z + z^T M z,\ x, z \in \mathbb{R}^n,
\end{aligned}$$

and so

$$h'\left(x\right) = x^T\left(M + M^T\right) + d^T,\ x \in \mathbb{R}^n,$$

and

$$h\left(x+z\right)=h\left(x\right)+h'\left(x\right)z+z^{T}Mz.$$

We have

$$\begin{aligned}
h\left(y_{n+1}\right) &= h\left(y_{n}+\sigma_{n+1}\sum_{i=1}^{\nu}b_{i}\frac{\kappa'\left(s_{i}^{n+1}\right)f\left(Y_{i}^{n+1}\right)}{h'\left(Y_{i}^{n+1}\right)f\left(Y_{i}^{n+1}\right)}\right)\\
&= h\left(y_{n}\right)+\sigma_{n+1}\sum_{i=1}^{\nu}b_{i}\frac{h'\left(y_{n}\right)\kappa'\left(s_{i}^{n+1}\right)f\left(Y_{i}^{n+1}\right)}{h'\left(Y_{i}^{n+1}\right)f\left(Y_{i}^{n+1}\right)}\\
&\quad +\sigma_{n+1}^{2}\sum_{i=1}^{\nu}\sum_{j=1}^{\nu}b_{i}b_{j}\frac{\kappa'\left(s_{i}^{n+1}\right)\kappa'\left(s_{j}^{n+1}\right)f\left(Y_{i}^{n+1}\right)^{T}Mf\left(Y_{j}^{n+1}\right)}{h'\left(Y_{i}^{n+1}\right)f\left(Y_{i}^{n+1}\right)\cdot h'\left(Y_{j}^{n+1}\right)f\left(Y_{j}^{n+1}\right)}.
\end{aligned}$$

Now, for $i=1,\ldots,\nu$,

$$\begin{aligned}
h'\left(y_{n}\right) &= y_{n}^{T}\left(M+M^{T}\right)+d^{T}\\
&= \left(Y_{i}^{n+1}-\sigma_{n+1}\sum_{j=1}^{\nu}a_{ij}\frac{\kappa'\left(s_{j}^{n+1}\right)f\left(Y_{j}^{n+1}\right)}{h'\left(Y_{j}^{n+1}\right)f\left(Y_{j}^{n+1}\right)}\right)^{T}\left(M+M^{T}\right)+d^{T}\\
&= h'\left(Y_{i}^{n+1}\right)-\sigma_{n+1}\sum_{j=1}^{\nu}a_{ij}\frac{\kappa'\left(s_{j}^{n+1}\right)f\left(Y_{j}^{n+1}\right)^{T}\left(M+M^{T}\right)}{h'\left(Y_{j}^{n+1}\right)f\left(Y_{j}^{n+1}\right)}
\end{aligned}$$

and then

$$\begin{aligned}
&\sigma_{n+1}\sum_{i=1}^{\nu} b_i \frac{\kappa'\left(s_i^{n+1}\right)h'\left(y_{n+1}\right)f\left(Y_i^{n+1}\right)}{h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)}\\
&=\sigma_{n+1}\sum_{i=1}^{\nu} b_i \frac{\kappa'\left(s_i^{n+1}\right)h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)}{h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)}\\
&\quad -\sigma_{n+1}^2\sum_{i=1}^{\nu}\sum_{j=1}^{\nu} b_i a_{ij}\frac{\kappa'\left(s_i^{n+1}\right)\kappa'\left(s_j^{n+1}\right)f\left(Y_j^{n+1}\right)^T Mf\left(Y_i^{n+1}\right)}{h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)\cdot h'\left(Y_j^{n+1}\right)f\left(Y_j^{n+1}\right)}\\
&\quad -\sigma_{n+1}^2\sum_{i=1}^{\nu}\sum_{j=1}^{\nu} b_i a_{ij}\frac{\kappa'\left(s_i^{n+1}\right)\kappa'\left(s_j^{n+1}\right)f\left(Y_j^{n+1}\right)^T M^T f\left(Y_i^{n+1}\right)}{h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)\cdot h'\left(Y_j^{n+1}\right)f\left(Y_j^{n+1}\right)}\\
&=\sigma_{n+1}\sum_{i=1}^{\nu} b_i\kappa'\left(s_i^{n+1}\right)\\
&\quad -\sigma_{n+1}^2\sum_{j=1}^{\nu}\sum_{i=1}^{\nu} b_j a_{ji}\frac{\kappa'\left(s_j^{n+1}\right)\kappa'\left(s_i^{n+1}\right)f\left(Y_i^{n+1}\right)^T Mf\left(Y_j^{n+1}\right)}{h'\left(Y_j^{n+1}\right)f\left(Y_j^{n+1}\right)\cdot h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)}\\
&\quad -\sigma_{n+1}^2\sum_{i=1}^{\nu}\sum_{j=1}^{\nu} b_i a_{ij}\frac{\kappa'\left(s_i^{n+1}\right)\kappa'\left(s_j^{n+1}\right)f\left(Y_i^{n+1}\right)^T Mf\left(Y_j^{n+1}\right)}{h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)\cdot h'\left(Y_j^{n+1}\right)f\left(Y_j^{n+1}\right)}\\
&=\sigma_{n+1}\sum_{i=1}^{\nu} b_i\kappa'\left(s_i^{n+1}\right)\\
&\quad -\sigma_{n+1}^2\sum_{i=1}^{\nu}\sum_{j=1}^{\nu}\left(b_i a_{ij}+b_j a_{ji}\right)\frac{\kappa'\left(s_i^{n+1}\right)\kappa'\left(s_j^{n+1}\right)f\left(Y_i^{n+1}\right)^T Mf\left(Y_j^{n+1}\right)}{h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)\cdot h'\left(Y_j^{n+1}\right)f\left(Y_j^{n+1}\right)}.
\end{aligned}$$

We conclude

$$\begin{aligned}
h\left(y_{n+1}\right)=h\left(y_n\right)+\sigma_{n+1}\sum_{i=1}^{\nu} b_i\kappa'\left(s_i^{n+1}\right)&\\
+\sigma_{n+1}^2\sum_{i=1}^{\nu}\sum_{j=1}^{\nu}\left(b_i b_j - b_i a_{ij} - b_j a_{ji}\right)&\frac{\kappa'\left(s_i^{n+1}\right)\kappa'\left(s_j^{n+1}\right)f\left(Y_i^{n+1}\right)^T Mf\left(Y_j^{n+1}\right)}{h'\left(Y_i^{n+1}\right)f\left(Y_i^{n+1}\right)\cdot h'\left(Y_j^{n+1}\right)f\left(Y_j^{n+1}\right)}
\end{aligned}$$

and if the quadrature rule $\left(b_i, c_i\right)_{i=1,\ldots,\nu}$ has polynomial order $m-1$, then

$$\sigma_{n+1}\sum_{i=1}^{\nu} b_i\kappa'\left(s_i^{n+1}\right)=\kappa\left(s_{n+1}\right)-\kappa\left(s_n\right).$$

■

Observe that we have

$$h\left(y_{n+1}\right)=h\left(y_n\right)+\kappa\left(s_{n+1}\right)-\kappa\left(s_n\right),\ n=0,1,2,\ldots,$$

if

$$b_i b_j - b_i a_{ij} - b_j a_{ji} = 0,\ i,j = 1,\dots,\nu,$$

holds. This is the condition for preserving quadratic first integrals and it cannot be satisfied if (A,b,c) is an explicit RK method (A,b,c). On the other hand, it is satisfied if (A,b,c) is a gaussian RK method.

In case of h quadratic, both in procedure A and procedure B, an integration of the transformed equation by a gaussian RK method guarantees exact numerical landing on Σ.

As for the one-sided integration, we can give the following interesting result for the procedure B with $\kappa(s) = s$.

Theorem 8 *Assume that h is quadratic and that in the procedure* B *with $\kappa(s) = s$ the transformed equation is integrated over one step by a RK method (A,b,c) such that*

$$\sum_{j=1}^{\nu} a_{ij} = c_i,\ i = 1,\dots,\nu,$$

Then, for $i = 1,\dots,\nu$, we have

$$h(Y_i^1) = h(x_{\overline{n}})\left(1 - c_i + O(\tau)\right),\ \tau \to 0.$$

Proof. Similarly to the proof of the previous theorem, we can show, for $i = 1,\dots,\nu$, that

$$\begin{aligned} h\left(Y_i^1\right) &= h(y_0) + \sigma_1 \sum_{j=1}^{\nu} a_{ij} \\ &\quad + \sigma_1^2 \sum_{j=1}^{\nu}\sum_{k=1}^{\nu} \left(a_{ij}a_{ik} - a_{ij}a_{jk} - a_{ik}a_{kj}\right) \frac{f\left(Y_j^1\right)^T M f\left(Y_k^1\right)}{h'\left(Y_j^1\right) f\left(Y_j^1\right) \cdot h'\left(Y_k^1\right) f\left(Y_k^1\right)}. \end{aligned}$$

Thus, since $\sigma_1 = -h(y_0) = -h(x_{\overline{n}})$ we have

$$\begin{aligned} h\left(Y_i^1\right) &= (1 - c_i) h(x_{\overline{n}}) \\ &\quad + h\left(x_{\overline{n}}\right)^2 \sum_{j=1}^{\nu}\sum_{k=1}^{\nu} \left(a_{ij}a_{ik} - a_{ij}a_{jk} - a_{ik}a_{kj}\right) \frac{f\left(Y_j^1\right)^T M f\left(Y_k^1\right)}{h'\left(Y_j^1\right) f\left(Y_j^1\right) \cdot h'\left(Y_k^1\right) f\left(Y_k^1\right)} \\ &= h(x_{\overline{n}})\left(1 - c_i + O\left(h(x_{\overline{n}})\right)\right) \\ &= h(x_{\overline{n}})\left(1 - c_i + O(\tau)\right),\ \tau \to 0, \end{aligned}$$

by recalling Lemma 3. ■

So, for $i = 1,\dots,\nu$ such that $c_i \neq 1$ and small τ, we have

$$h\left(Y_i^1\right) = h(x_{\overline{n}})\left(1 - c_i + O(\tau)\right) = (1 - c_i)\, h\left(x_{\overline{n}}\right)\left(1 + O(\tau)\right) \approx (1 - c_i)\, h\left(x_{\overline{n}}\right) < 0.$$

In case of h quadratic, an one-sided integration in the procedure B can be indeed realized by using an explicit RK method with $c_1,\dots,c_\nu < 1$.

Example 9 *Consider the problem with* h *quadratic*

$$f\left(x\right)=\left(x_2,-x_1+\frac{1}{1.2-x_2}\right),\ t_0=0,\ x_0=\left(-0,2,-0.2\right),$$
$$h\left(x\right)=x_1^2+x_2^2+x_1+x_2-0.4.$$

In the procedure B*, we integrate the original equation by the Heun method and the transformed equation with* $\kappa\left(s\right)=s$ *by the explicit midpoint method, whose tableau is*

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & 1 \end{array}.$$

We obtain the table

τ	$\lvert\varepsilon_2\rvert$	*ratios*
10^{-1}	$9.94\cdot10^{-3}$	4.50
10^{-2}	$2.21\cdot10^{-3}$	7.85
10^{-3}	$2.81\cdot10^{-4}$	20.9
10^{-4}	$1.35\cdot10^{-5}$	6.75
10^{-5}	$1.99\cdot10^{-6}$	

The relative error $\varepsilon_2=\frac{h(Y_2^1)-(1-c_2)h(x_{\overline{n}})}{(1-c_2)h(x_{\overline{n}})}$ *exhibits order* $O(\tau)$ *(the geometric mean of the ratios is* 8.40*).*

5 Tangential landing on Σ

Now we consider the situation of *tangential landing* on the border Σ, i.e. we have

$$h'\left(x\right)f\left(x\right)>0,\ x\in R \text{ such that } h(x)>-c,$$

instead of (8), and

$$h'(x(t_f))f(x(t_f))=0.$$

In the procedure A, assume to use $\kappa(s)=s$ in the transformed equation (9). Since in a tangential landing on Σ the quantity $h'\left(y\left(s\right)\right)f\left(y\left(s\right)\right)$ becomes zero as s approaches zero, a numerical method integrating the transformed equation (9) encounters difficulties due to the blowing up right-hand side

$$\frac{\kappa'(s)}{h'\left(y\left(s\right)\right)f\left(y\left(s\right)\right)}\left[\begin{array}{c} f\left(y\left(s\right)\right) \\ 1 \end{array}\right]=\frac{1}{h'\left(y\left(s\right)\right)f\left(y\left(s\right)\right)}\left[\begin{array}{c} f\left(y\left(s\right)\right) \\ 1 \end{array}\right].$$

Clearly, the situation of tangential landing on Σ is one where a function κ different from the simplest choice $\kappa(s)=s$ should be used.

Let us define

$$\beta\left(t\right)=h(x(t)),\ t\in\left[0,t_f\right],$$

and then

$$\beta'\left(t\right)=h'(x(t))f\left(x\left(t\right)\right),\ t\in\left[0,t_f\right].$$

The situation of tangential landing on Σ is characterized by

$$\beta\left(t_f\right) = \beta'\left(t_f\right) = 0.$$

Proposition 10 *Let k be the positive integer such that*

$$\beta^{(i)}\left(t_f\right) = 0,\ i = 0, 1, \ldots, k, \quad \textit{and} \quad \beta^{(k+1)}\left(t_f\right) \neq 0. \tag{18}$$

If

$$\frac{\kappa'(s)}{\left|\kappa\left(s\right)\right|^{\frac{k}{k+1}}},\ s \in [s_0, 0], \tag{19}$$

is a smooth function, then

$$\frac{\kappa'(s)}{h'(y(s))f\left(y\left(s\right)\right)},\ s \in [s_0, 0],$$

in (9) is a smooth function.

Proof. We have

$$\left|\beta\left(t\right)\right| = a\left(t_f - t\right)^{k+1} \cdot \left(1 + R(t)\right), t \in [t_0, t_f],$$

where

$$a = (-1)^k \frac{\beta^{(k+1)}\left(t_f\right)}{(k+1)!}$$

and R is a smooth function such that

$$R(t) \to 0,\ t \to t_f.$$

Moreover

$$\beta'\left(t\right) = a_1\left(t - t_f\right)^k \cdot \left(1 + R_1(t)\right),\ t \in [t_0, t_f],$$

where

$$a_1 = (-1)^k \frac{\beta^{(k+1)}\left(t_f\right)}{k!}$$

and R_1 is a smooth function such that

$$R_1(t) \to 0,\ t \to t_f.$$

Thus, we have

$$\begin{aligned}
\alpha'(s) &= \frac{\kappa'(s)}{h'(y(s))f\left(y\left(s\right)\right)} = \frac{\kappa'(s)}{\beta'(\alpha(s))} \\
&= \frac{\kappa'(s)}{\left|\kappa\left(s\right)\right|^{\frac{k}{k+1}}} \cdot \frac{1}{\frac{a_1(\alpha(s)-t_f)^k \cdot (1+R_1(\alpha(s)))}{|\beta(\alpha(s))|^{\frac{k}{k+1}}}} \quad \text{since} \ \ \kappa(s) = \beta(\alpha(s)) \\
&= \frac{\kappa'(s)}{\left|\kappa\left(s\right)\right|^{\frac{k}{k+1}}} \cdot \frac{1}{\frac{a_1 \cdot (1+R_1(\alpha(s)))}{a^{\frac{k}{k+1}}(1+R(\alpha(t))^{\frac{k}{k+1}}}},\ s \in [s_0, 0]
\end{aligned}$$

If (19) is a smooth function, then also the solution α of this differential equation is a smooth function and so

$$\frac{\kappa'(s)}{h'(y(s))f\left(y\left(s\right)\right)} = \alpha'(s),\ s \in [s_0, 0],$$

is a smooth function. ■

In case of functions κ of type $\kappa_{m,C}$ given in (11), we obtain

$$\frac{\kappa'_{m,C}\left(s\right)}{\left|\kappa_{m,C}\left(s\right)\right|^{\frac{k}{k+1}}} = \frac{mC\left(-s\right)^{m-1}}{C^{\frac{k}{k+1}}\left(-s\right)^{\frac{k}{k+1}m}} = mC^{\frac{1}{k+1}}\left(-s\right)^{\frac{m}{k+1}-1},\ s \in [s_0, 0],$$

By taking a function $\kappa_{m,C}$ with $m \geq k+1$, we can avoid the blow-up in (9) and have a smooth solution of (9).

Next example shows that a function κ different from the simplest choice $\kappa\left(s\right) = s$ can work better in case of tangential landing.

Example 11 *Consider the problem*

$$\begin{aligned} &f\left(x\right) = Ax,\ t_0 = 0,\ x_0 = \mathrm{e}^{-Aa}, \\ &h\left(x\right) = x_1 + x_2 - 3, \end{aligned}$$

where

$$A = \left[\begin{array}{cc} 1 & 1 \\ -2 & 1 \end{array}\right],\ a = (2, 1).$$

The solution is

$$x\left(t\right) = \mathrm{e}^{tA},\ t \geq 0,$$

and the event

$$h(x(t)) = 0$$

is located at $t_f = 1$ with value $x\left(t_f\right) = a$.

We have $\beta\left(t\right) = h\left(x\left(t\right)\right)$ with

$$\beta\left(0\right) = 0,\ \beta'\left(0\right) = 0,\ \beta''\left(0\right) = -9.$$

So, we are in the situation of tangential landing. In Figure 2, we see the trajectory of the solution in the phase space.

In the procedure A, *we integrate by the Heun method the transformed equation with $\kappa\left(s\right) = s$, $\kappa\left(s\right) = -s^2$ and $\kappa(s) = s^3$. We obtain the following errors*

$$|\alpha_N - t_f|, \|\ y_N - x\left(t_f\right)\|_\infty$$

for a constant stepsize $\sigma = 10^{-k}$, $k = 1, \ldots, 4$, where t_f and $x\left(t_f\right)$ are exactly known and α_N and y_N are their numerical approximations.

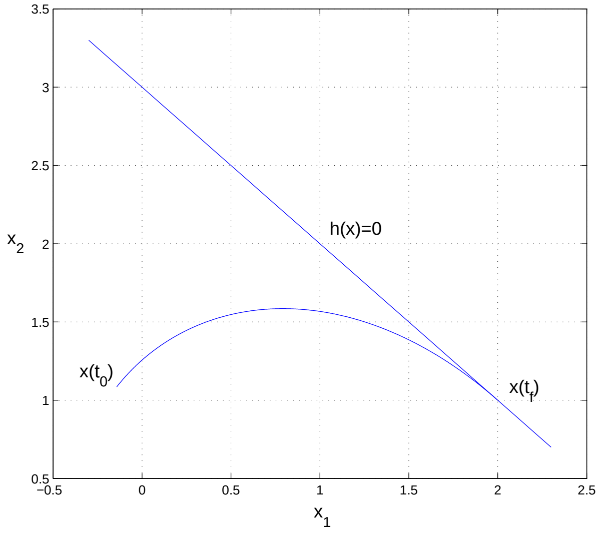

Figure 2: The trajectory $x(t)$ in the phase space.

For $\kappa\left(s\right)=s$*:*

τ	$\left\|\alpha_N-t_f\right\|$	*ratios*	$\left\|y_N-x\left(t_f\right)\right\|_\infty$	*ratios*
10^{-1}	$8.24\cdot10^{-2}$	4.19	$2.38\cdot10^{-1}$	4.05
10^{-2}	$1.97\cdot10^{-2}$	3.46	$5.88\cdot10^{-2}$	3.44
10^{-3}	$5.69\cdot10^{-3}$	3.24	$1.71\cdot10^{-3}$	3.24
10^{-4}	$1.76\cdot10^{-3}$	3.19	$5.26\cdot10^{-3}$	3.18
10^{-5}	$5.51\cdot10^{-4}$		$1.65\cdot10^{-3}$	

For $\kappa\left(s\right)=-s^2$*:*

τ	$\left\|\alpha_N-t_f\right\|$	*ratios*	$\left\|y_N-x\left(t_f\right)\right\|_\infty$	*ratios*
10^{-1}	$8.95\cdot10^{-2}$	9.53	$2.57\cdot10^{-1}$	9.18
10^{-2}	$9.39\cdot10^{-3}$	9.98	$2.80\cdot10^{-2}$	9.92
10^{-3}	$9.41\cdot10^{-4}$	10.0	$2.82\cdot10^{-3}$	9.99
10^{-4}	$9.41\cdot10^{-5}$	10.1	$2.82\cdot10^{-4}$	10.1
10^{-5}	$9.36\cdot10^{-6}$		$2.81\cdot10^{-5}$	

For $\kappa\left(s\right)=s^3$*:*

τ	$\left\|\alpha_N-t_f\right\|$	*ratios*	$\left\|y_N-x\left(t_f\right)\right\|_\infty$	*ratios*
10^{-1}	$1.07\cdot10^{-1}$	9.45	$3.16\cdot10^{-1}$	9.35
10^{-2}	$1.13\cdot10^{-2}$	10.0	$3.38\cdot10^{-2}$	9.99
10^{-3}	$1.13\cdot10^{-3}$	10.0	$3.38\cdot10^{-3}$	10.0
10^{-4}	$1.13\cdot10^{-4}$	9.97	$3.38\cdot10^{-4}$	9.97
10^{-5}	$1.13\cdot10^{-5}$		$3.39\cdot10^{-5}$	

Thus, by replacing the simplest choice $\kappa(s) = s$ with the quadratic function $\kappa(s) = -s^2$ or the cubic function $\kappa(s) = s^3$, we can improve the order of convergence from one half to one, although we do not reach the order two of the non-tangential situation.

Now, we try to explain why there is such an order reduction for the Heun method in the tangential situation. We show that the usual argument used for proving that the Heun method has convergence order two fails in this situation.

Consider the Heun method as applied to a general equation

$$y'(s) = G(s, y(s)), s \in [s_0, 0],$$

with stepsize σ. As usual, we split the error $y_{n+1} - y(s_{n+1})$ at the $(n+1)$-th step as

$$y_{n+1} - y(s_{n+1}) = y_{n+1} - z_{n+1} + z_{n+1} - y(s_{n+1}),$$

where z_{n+1} is the numerical solution at the $(n+1)$-th step when we replace y_n with $y(s_n)$. Fixed an arbitrary $\varepsilon > 0$, the local error $z_{n+1} - y(s_{n+1})$ can be bounded by

$$\|z_{n+1} - y(s_{n+1})\| \leq \left(\frac{5}{12} \max_{s \in [s_n, s_{n+1}]} \|y'''(s)\| + \frac{1}{4} L_{n+1} \max_{s \in [s_n, s_{n+1}]} \|y''(s)\| \right) \sigma^3,$$

where

$$L_{n+1} = \max_{s \in [s_n, s_{n+1}] \text{ and } \|w\| \leq \varepsilon} \left\| \frac{\partial G}{\partial z}(s, y(s) + w) \right\|,$$

whenever

$$\frac{1}{2} \max_{s \in [s_n, s_{n+1}]} \|y''(s)\| \sigma^2 \leq \varepsilon.$$

Moreover, the propagated error $y_{n+1} - z_{n+1}$ can be bounded by

$$\|y_{n+1} - z_{n+1}\| \leq \left(1 + \tau L_{n+1} + \frac{(\tau L_{n+1})^2}{2} \right) \|y_n - y(s_n)\|$$

whenever

$$(1 + \tau L_{n+1}) \|y_n - y(s_n)\| \leq \varepsilon.$$

Now, we consider the transformed equation

$$y'(s) = \frac{\kappa'(s)}{h' f(y(s))} f(y(s)), \ s \in [s_0, 0],$$

with h' constant, i.e. the case h linear, as in the previous example. Here, we have

$$G(s, y) = \frac{\kappa'(s)}{h' f(y)} f(y), \ s \in [s_0, 0] \text{ and } y \in \mathbb{R}^n,$$

and so

$$\frac{\partial G}{\partial y}(s, y)p = \frac{\kappa'(s)}{h' f(y)} f'(y) - \frac{h' f'(y)p}{h' f(y)} G(s, y), \ s \in [s_0, 0] \text{ and } y, p \in \mathbb{R}^n.$$

According to Proposition 10, the use of a suitable function κ guarantees that the terms involving the second and third derivatives of the solution in the bound of the local error do not blow up as s_n approaches zero. On the other hand, we have

$$L_{n+1} \approx \max_{s\in[s_n,s_{n+1}]} \left\| \frac{\partial G}{\partial z}(s,y(s)) \right\|$$

with

$$\frac{\partial G}{\partial z}(s,y(s))p = \frac{\kappa'(s)}{h'f(y(s))}f'(y(s)) - \frac{h'f'(y(s))p}{h'f(y(s))}G(s,y(s)),\ s \in [s_0,0] \text{ and } p \in \mathbb{R}^n.$$

Observe that the denominator $h'f(y(s))$ tends to zero as $s \to 0$, but, unlike the first term, this cannot be controlled by a suitable function κ in the second term. Therefore, L_{n+1} blows up as s_n approaches zero. This explains why the full order two of the Heun method fails in the tangential situation.

However, we remark that, although the use of a function κ different from the simplest choice $\kappa(s) = s$ cannot recover the full order two of the Heun method, it can be recommended because anyway it improves the order of convergence as it is shown in the previous example. The study of these reduced orders appearing in the tangential situation will be addressed in a next paper.

6 Conclusion

In this paper, we have presented an approach for the location of events for ordinary differential equations based on suitable transformations of the independent variable time called time-transformations. The approach is implemented in two procedures called A and B. The procedure A is a generalization of a method proposed in [12] and this generalization permits to deal better critical situations as in case of a solution reaching the event in a tangential way. On the other hand, the procedure B should be used in the non-tangential case and it is the neatest and most efficient manner to use the time-transformations since they are used only when they become necessary.

Acknowledgements. The paper has been supported in part by the GNCS of the Italian “Istituto Nazionale di Alta Matematica”.

References

[1] V. Acary and B. Brogliato. Numerical Methods for Nonsmooth Dynamical Systems. Applications in Mechanics and Electronics. Lecture Notes in Applied and Computational Mechanics. Springer-Verlag, Berlin, 2008.

[2] A. Bellen and M. Zennaro. Numerical Methods for Delay Differential Equations. Numerical Mathemathics and Scientific Computations. Clarendon Press, Oxford 2003.

[3] A. Berardi and L. Lopez On the continuous extension of Adams Bashforth methods and the event location in discontinuous ODEs. Applied Mathematics Letters, vol. 25, (6), pp. 995-999, 2012.

[4] H. Brunner and S. Maset. Time transformations for delay differential equations. Discrete and Continuous Dynamical Systems, vol. 25 (3), pp. 751-775, 2009.

[5] H. Brunner and S. Maset. Time transformations for state dependent delay differential equations. Communication on Pure and applied Analysis, vol. 9 (1), pp. 23-45, 2010.

[6] M. Calvo, J.L. Montijano, and L. Randez. On the solution of discontinuous IVPs by adaptive Runge-Kutta codes. Numerical Algorithms, vol. 33 (1), pp. 163-182, 2003.

[7] M. Calvo, J.I. Montijano, and L. Randez. DISODE45: A Matlab Runge-Kutta solver for Piecewise Smooth IVPs of Filippov type. ACM Transactions on Mathematical Software (TOMS) Vol. 43, Issue 3, January 2017, Art. No. 25 , 2016.

[8] N. Del Buono, C. Elia and L. Lopez. On the equivalence between the sigmoidal approach and Utkin's approach for piecewise-linear models of gene regulatory networks. SIAM Journal on Applied Dynamical Systems, vol. 13 (3), pp. 1270-1292, 2014.

[9] N. Del Buono and L. Lopez. Direct event location techniques based on Adams multistep methods for discontinuous ODEs. Applied Mathematics Letters, vol. 49, pp. 152-158, 2015.

[10] M. di Bernardo, C. Budd, A.R. Champneys, P. Kowalczyk. Piecewise-smooth dynamical systems: theory and applications. Springer Science & Business Media, Berlin, 2008.

[11] L. Dieci and L. Lopez. Numerical solution of discontinuous differential systems: Approaching the discontinuity surface from one side. Applied Numerical Mathematics, vol. 67, pp. 98-110, 2013.

[12] L. Dieci and L. Lopez. One-Sided Direct Event Location Techniques in the Numerical Solution of Discontinuous Differential Systems. BIT Numerical Mathematics, vol. 55 (4), pp. 987-1003, 2015.

[13] J. M. Esposito, V. Kumar, and G.J. Pappas. Accurate Event Detection for Simulating Hybrid Systems. M.D. Di Benedetto, A. Sangiovanni-Vincentelli (Eds.): HSCC 2001, LNCS 2034, pp. 204- 217, Springer-Verlag Berlin Heidelberg, 2001.

[14] N. Guglielmi and E. Hairer. Computing breaking points in implicit delay differential equations. Advances in Computational Mathematics, vol. 29, pp. 229–247, 2008.

[15] N. Guglielmi and E. Hairer. Classification of Hidden Dynamics in Discontinuous Dynamical Systems. SIAM J. Appl. Dyn. Syst., vol. 14 (3), pp. 1454-1477, 2015.

[16] B. Kacewicz and P. Przybylowicz. Complexity of the derivative-free solution of systems of IVPs with unknown singularity hypersurface. Journal of Complexity, vol. 31 (1) , pp. 75-97, 2015.

[17] B. Kacewicz and P. Przybylowicz. Optimal solution of a class of non-autonomous initial-value problems with unknown singularities. Journal of Computational and Applied Mathematics, vol. 261, pp. 364-377, 2014.

[18] L.F. Shampine and S. Thompson. Event location for ordinary differential equations. Computer and Mathematics with Applications, vol. 39, pp. 43–54, 2000.