International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:5, No:12, 2011

Ordinary differential equations with inverted functions

Thomas Kämpke

Abstract—Equations with differentials relating to the inverse of an unknown function rather than to the unknown function itself are solved exactly for some special cases and numerically for the general case. Invertibility combined with differentiability over connected domains forces solutions always to be monotone. Numerical function inversion is key to all solution algorithms which either are of a forward type or a fixed point type considering whole approximate solution functions in each iteration. The given considerations are restricted to ordinary differential equations with inverted functions (ODEIs) of first order. Forward type computations, if applicable, admit consistency of order one and, under an additional accuracy condition, convergence of order one.

Keywords—Euler method, fixed points, golden section, multistep procedures, Runge Kutta methods.

I. Introduction

DIFFERENTIAL equations come in a plethora of types, but in a far less common line of modifications they allow to be altered so that they involve the derivative and the inverse of some unknown function. But the unknown function itself is not a component of the equation. So modifying one of the simplest ordinary differential equations, the linear homogenous equation with one constant coefficient, leads to the differential equation

$$y' = c \cdot y^{-1}.$$

The constant c is assumed to be positive and the unknown function y is replaced by its inverse function y^{-1}. The equation describes invariants of a certain transformation of Lorenz curves [4] but it can be considered for its own sake when invertibility of the unknown function can be assumed. A closed-form solution is

$$y(x) = \left(\frac{c}{g}\right)^{1/g} \cdot x^g$$

with golden section ratio $g = (1+\sqrt{5})/2$. The solution can be found by observing that the type of a power function is preserved by inversion and differentiation. This motivates to consider functions $y(x) = k \cdot x^a$. Insertion into the differential equation and comparing exponents results in $a = g$ and, subsequently, comparing the constant values on both sides of the remaining equation allows to resolve for the factor k. The same approach applies to the slightly more general differential equations

$$y' = c \cdot \left(y^{-1}\right)^n, \ n = 1, 2, \ldots$$

T. Kämpke is with the Research Institute of Applied Knowledge Processing FAW/n, Ulm, Germany e-mail: (see http://www.faw-neu-ulm.de).

with $c > 0$. Solutions are

$$y(x) = \left(\frac{c}{g_n}\right)^{1/g_n} \cdot x^{g_n}$$

with exponents g_n being the positive roots of the quadratic equation $a^2 - a - n = 0$. The sequence of these solutions generalizes the golden section and has the first three values

$$\begin{aligned} g_1 &= \frac{1+\sqrt{5}}{2}; = g \\ g_2 &= \frac{1+\sqrt{9}}{2} = 2 \\ g_3 &= \frac{1+\sqrt{13}}{2}. \end{aligned}$$

This sequence must not be confused with the so-called metallic means family that generalizes the golden section as the sequence of the golden, silver, bronze section etc. which are the positive roots of $a^2 - n\,a - 1 = 0$, see [3].

In the following, the focus will be on more systematic solution strategies. Also, it is to be found out if other solutions of the introductory differential equation exist and what effects initial boundary values can have. The general form of differential equations considered here is that of a general ordinary differential equation but with the unknown function – not its derivative – replaced by its inverse. With a real-valued and continuous function F, denoted as structure function or system function, the general form of an ordinary differential equation of first kind with inverted function is spelled out as

$$y'(x) = F(y^{-1}(x), x).$$

A differential equation with inverted function is abbreviated as an ODEI and some ODEIs which admit closed-form solutions are summarized in table 1. ODEIs lack linear solution spaces and straightforward, symbolic integration so that ”most” ODEIs will require numerical solutions.

International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:5, No:12, 2011

Solution function/ ODEI
$y(x) = x^a$ (power functions)
$y'(x) = y^{-1}(x) \cdot a \cdot x^{a-1-1/a},\ a > 1$
$y(x) = x^a$ (power functions)
$y'(x) = a \cdot (y^{-1}(x))^{a(a-1)},\ a > 1$
$y(x) = 1/x$ (hyperbolic function)
$y'(x) = -(y^{-1}(x))^2$
$y(x) = 1/x^2$ (hyperbolic function)
$y'(x) = -2 \cdot (y^{-1}(x))^6$
$y(x) = \ln(x)$
$y'(x) = 1/\ln(y^{-1}(x))$
$y(x) = exp(x)$
$y'(x) = exp(exp(y^{-1}(x)))$

Table 1: Sample ODEIs and their solution functions.

The system functions can be replaced by more general operators including those for time-delayed systems [2] and including integration. An example of the latter is $y'(x) = \int_0^x y^{-1}(u)\, du$ which has the solution function $y(x) = (1/(s+1))^{s/(s+1)} \cdot x^s$ with silver section $s = 1 + \sqrt{2}$. These more general kinds of differential equations are not considered here but, obviously, integration allows to rewrite ODEIs which will be exploited in Runge Kutta-like numerical solutions.

The remainder of the paper is organized as follows. Some principal facts preparing for numerical computing schemes are stated in section 2. Focus is laid on a reformulation of ODEIs as ODEs with function concatenation and on an approximation of the inverse function based on some approximation of the solution function itself.

The difference between ordinary differential equations with mere functions and those with inverses or concatenation becomes obvious by some solution concepts being no longer or not directly applicable. One such concept is the direction field. Also, existence results are difficult to obtain as solvability of differential equations is, generally, only guaranteed locally. But a small domain may be too small to define the inverse or the concatenation of a function there. Thus, fixed point theorems that often guarantee solvability of ODEs become, here, difficult if not impossible to apply.

Section 3 contains iterative fixed point methods as well as various forward methods that apply in special cases or under special provisions. The obvious difficulty that sets all these methods apart from solving ordinary differential equations is that the inverse function has to be computed in some form or another. This will be achieved ether separately or interleaved with computations of the solution function.

Forward methods as well as iterations with complete functions will be explored and their consistency as well as their convergence will be analyzed. However, power series method will not be inspected since the Taylor series of a differentiable function and its inverse may have significantly different convergence radii [5].

Bounds on solution functions as well as a few approximations of closed-form type are stated in the final section 4.

II. Structure

A. Linear and quadratic closed-form solutions

Beyond the cursory examples from the introduction, some special classes of ordinary differential equations with inverted function are solvable in closed form. While the ODEI $y'(x) = D^2 \cdot y^{-1}(x)/(x - A)$ with $D \neq 0$ has the linear solution functions $y(x) = D \cdot x + A$ for $x > -A$, the quadratic ODEI $y' = A \cdot (y^{-1} + B)^2 + C$ with constants $A > 0$, B and C has the solution functions and solution coefficients

$$y(x) = a\,x^2 + b\,x + c$$

$$a = \sqrt{\frac{A}{2}},\ b = \sqrt{2\,A}\,B,\ c = \frac{C}{\sqrt{2A}} + \frac{\sqrt{A}\,B^2}{\sqrt{2}} - B.$$

These solutions are valid for sufficiently large arguments and condition $b^2 - 4\,a\,c \geq 0$ suffices for the solutions to be valid over all non-negative reals. No parameter constellation admits the quadratic functions to become, in particular, linear.

B. Reformulations

Whenever possible, solution functions are assumed to be strictly increasing. Then the original differential equation with constant coefficient $y' = c \cdot y^{-1}$ has no solution with value $y(0) > 0$ over the non-negative reals. Assuming it had a solution with such a value and assuming that the inverse function were defined at zero implies, by reflection along the diagonal line $y = x$, that the inverse function has a zero at $y(0) > 0$. Then, because the inverse is strictly increasing like the solution itself, $y^{-1}(0) < y^{-1}(y(0)) = 0$. Since $y'(0) = c \cdot y^{-1}(0) < 0$, the solution function were strictly decreasing near zero, a contradiction.

The same result can be obtained for the differential equation in general form if the system function is strictly increasing in the first argument and $F(0,0) \leq 0$. When, in addition, the system function is increasing in both arguments, each solution function being increasing then implies that it also is convex, since its derivative is increasing. The set of solution functions cannot be homogenous since, for $\alpha > 1$,

$$\begin{aligned}(\alpha \cdot y)'(x) &= \alpha \cdot y'(x) > y'(x) = F(y^{-1}(x), x)\\ &> F(1/\alpha \cdot y^{-1}(x), x) = F((\alpha \cdot y)^{-1}(x), x).\end{aligned}$$

The general ODEI can be reformulated equivalently without inverted function as $y'(y(x)) = F(x, y(x))$ by the substitution $x = y(u)$ and by naming the independent variable again as x. Also, the inverse of any solution of the ODEI satisfies the differential equation

$$(y^{-1})'(x) = \frac{1}{y'(y^{-1}(x))} = \frac{1}{F(y^{-1}(y^{-1}(x)), y^{-1}(x))}.$$

Renaming the inverse function again $y(x)$ and using another system function results in the ordinary differential equation with concatenation

$$y'(x) = f(y \circ y(x), y(x)).$$

Though explicit inverse functions are avoided in both reformulations, these forms of the ODEI do not seem to lead to any "standard" ODE type. Considered by themselves, differential equations with concatenation admit non-invertible solutions but the range of a concatenated function always must lie in its domain. For example, when the system function vanishes somewhere on the diagonal, which means that $f(c,c) = 0$ for some value c, then the concatenated ODE admits the constant solution function $y(x) = c$. Yet, solution functions are supposed to be invertible here and ODEs with concatenation will be used, below, for forward computations.

ODEs with concatenation admit a geometrical interpretation in terms of so-called scaled subtangents which slightly modify Leibniz' original subtangents as sketched in figure 1. The original subtangent for the point x is the interval $[\lambda(x), x]$ so that the tangent slope equals $y'(x) = y(x)/(x - \lambda(x))$, see [11, p. xix]. The scaled subtangent is understood to be the interval $[\gamma(x), y(x)]$. The triangles over both subtangents being similar allows to express the original slope, also, as $y'(x) = y(y(x))/(y(x) - \gamma(x))$. Requiring now that all subtangents and all scaled subtangents are constant, respectively, results in the linear and in the concatenated ODE

$$\begin{aligned} y'(x) &= const \cdot y(x) \\ y'(x) &= const \cdot y(y(x)). \end{aligned}$$

The requirements of all subtangents being constant and all scaled subtangents being constant differ so severely that solutions of the two differential equations are completely different. Yet particular solution functions may have common features such as both being either increasing and convex or decreasing and concave.

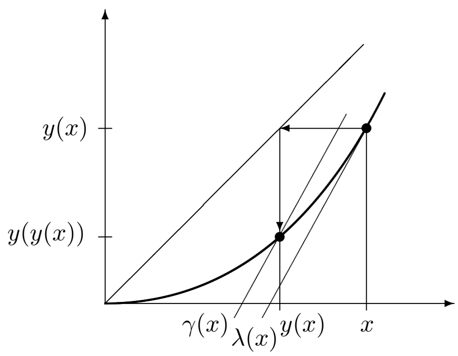

Fig. 1. Original subtangent $[\lambda(x), x]$ and scaled subtangent $[\gamma(x), y(x)]$.

A quite different reformulation of the ODEI results from a formula for integrating the composition of a system function and an inverse functions of the form $F(y^{-1}(x), x)$ [10]. The resulting differential equation neither requires inversion nor concatenation of the unknown function

$$\begin{aligned} y'(x) &= \left(\frac{d}{dx}F(x, y(x))\right) \cdot y(x) + F(x, y(x)) \cdot y'(x) \\ &\quad - y(x) \cdot \frac{d}{dx}F(x, y(x)). \end{aligned}$$

Though the integration formula as such may be convenient for known functions $y(x)$, the resulting differential equation hardly simplifies solution processes.

C. Solvability

Solutions of a concatenated ODE have a fixed point over a finite closed interval when the solution does not resort to arguments from the outside. This follows, since the concatenation of the continuous solution function is a mapping from its domain into its domain under the present circumstances. But over infinite domains, solutions of a concatenated ODE need not have a fixed point. An example is $y(x) = x + 1$ over $x > -1$; it satisfies the concatenated ODE $y'(x) = 1 = (x+2-1)/(x+1) = (y(y(x))-1)/y(x)$ with system function $f(a,b) = (a-1)/b$. On the other hand, a solution of the concatenated ODE with initial value $y(x_0) = y_0$ itself is a fixed point of the operator

$$T(y)(x) = y_0 + \int_{x_0}^{x} f(y \circ y(s), y(s))\, ds.$$

When the system function is bounded, the operator maps any continuous function to a Lipschitz-bounded function. Also, boundedness of the system function on some closed domain with support points $x_0 < x_1 < \ldots < x_n$ implies that an Euler polygon

$$p(x) = p(x_i) + (x - x_i)\, f(p(p(x_i)), p(x_i)),\ x_i \leq x \leq x_{i+1}$$

with initial condition $p(x_0) = y_0$ is Lipschitz-bounded when the concatenation does not leave that or another closed finite domain. However, Euler polygons for concatenated ODEs are not explicit as they are for mere ODEs since the approximations of all function values up to some support point do not suffice to approximate the next function value – concatenated function values must be known in addition.

A simple set of conditions for the concatenated ODEI having a local solution around a point x_0 is that an initial condition is satisfied as fixed point condition $y(x_0) = x_0$, the system function is Lipschitz-continuous there with Lipschitz constant less than one and $|f(x_0, x_0)| < 1$. As general existence results are not pivotal here, the argument is sketched only. Continuity of the system function implies the existence of a 2D interval $I = [x_0 - a, x_0 + a] \times [x_0 - b, x_0 + b]$ with $a, b > 0$ so that $M = max_{(x,y)\in I} |f(x,y)| < 1$. Selection of a some positive parameter $\alpha \leq \min\{1, a, b/M\}$ allows to consider the function space

$$\begin{aligned} \mathcal{F} &= \{y : [x_0 - \alpha, x_0 + \alpha] \\ &\quad \to [x_0 - M \cdot \alpha, x_0 + M \cdot \alpha] \,\Big|\, y \text{ is continuous}\}. \end{aligned}$$

The range of any function from $\mathcal{F}$ lies in its domain so that concatenation is feasible. The integral operator $T(y)(x) = x_0 + \int_{x_0}^{x} f(y(y(s)), y(s))\, ds$ maps any function from $\mathcal{F}$ to $\mathcal{F}$ because

$$\begin{aligned} |(Ty)(x) - x_0| &= \left| \int_{x_0}^{x} f(y(y(s)), y(s))\, ds \right| \\ &\leq M \cdot |x - x_0| \leq M \cdot \alpha. \end{aligned}$$

The Lipschitz condition $|f(u(u(s)), u(s)) - f(v(v(s)), v(s))| \leq L \cdot |u(s) - v(s)|$ with $L < 1$ implies that the integral operator is a contraction mapping on $\mathcal{F}$ since

$$\begin{aligned} |(Tu)(x)-(Tv)(x)| &= \Big| \int_{x_0}^{x} f(u(u(s)),u(s))\,ds \\ &\quad - \int_{x_0}^{x} f(v(v(s)),v(s))\,ds\,\Big| \\ &\leq |x-x_0|\cdot L\cdot ||u-v||_\infty \\ &\leq L\cdot ||u-v||_\infty \end{aligned}$$

with $||u-v||_\infty = max_{x\in[x_0-\alpha,x_0+\alpha]}|u(x)-v(x)|$. The Banach fixed point theorem for metric spaces [9, p. 2] then implies the existence of a fixed point in $\mathcal{F}$.

D. Approximations

An indispensible component of any solution method for ODEIs is some computation of the inverse. When an increasing solution function is approximated piecewise linearly and continuous, its inverse can also be approximated by a piecewise linear function. In the simplest case, the approximating segments of the inverse are formed by linear interpolation. This is possible, obviously, only for segments that are enclosed by support points.

The support points are chosen so that the differential equation is considered, at most, over the interval $[x_0, x_n]$. Let $y_0 < \ldots < y_n$ be approximate values of the solution function with $y_i \approx y(x_i)$ for all indices. Then the inverse solution function has the domain $[y(x_0), y(x_n)]$ and the range $[x_0, x_n]$. The ODEI is meaningful only over the intersection of $[x_0, x_n]$ – so that y' is defined – and $[y(x_0), y(x_n)]$ – so that y^{-1} is defined. The approximation domain is hence defined as

$$Dom = [x_0, x_n] \cap [y(x_0), y(x_n)].$$

The approximation domain may be empty in which case the considered domain or range of the solution function should be enlarged. If this is infeasible, the ODEI does not admit a solution. If the approximation domain is not empty, a discretized function can be tested to admit a linear interpolation of its inverse by the following procedure.

Test-and-invert

1) Find $j \in \{0, 1, \ldots, n-1\}$ with $y_j < x_i \leq y_{j+1}$.
2) Compute $y^{-1}(x_i) = x_j + \frac{x_i - y_j}{y_{j+1}-y_j}\cdot(x_{j+1}-x_j)$.

Due to the interpolation character of the procedure, candidate values of the inverse function can never be computed for the two boundary points of the function domain. In addition, an inverse value is not computed if no index j satisfies the sandwich property from step 1. Sample computations are given in figure 2; all actual computations and related drawings were done in Scilab [7]. The linear interpolation from step 2 is sketched in figure 3.

Computing inverse function values by interpolation can be improved in certain situations by applying splines rather linear segments. Slopes of the inverse function at the boundaries of intervals $[y_j, y_{j+1}]$ are related to certain values of the inverse

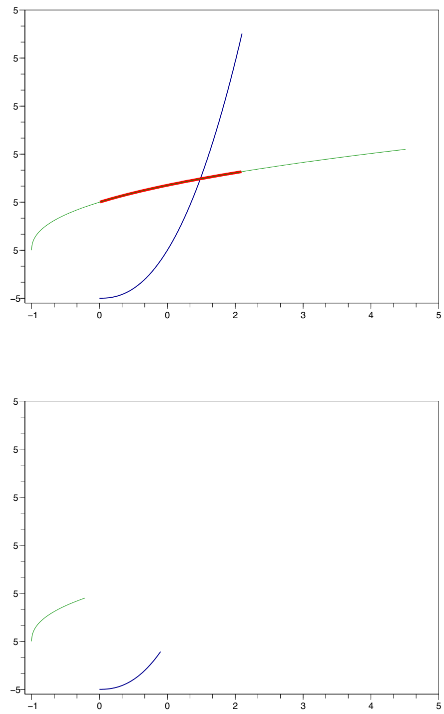

Fig. 2. Function $y(x) = x^{2.3} - 1$ over domain $[x_0, x_n] = [0, 2.1]$ (top) and $[x_0, x_n] = [0, 0.9]$ (bottom) with inverses. For the larger function domain, the inverse is also computed over the approximation domain by Test-and-invert (bold function segment). In the other case, the approximation domain is empty (right) since the largest value of the function is smaller than the smallest value of the inverse.

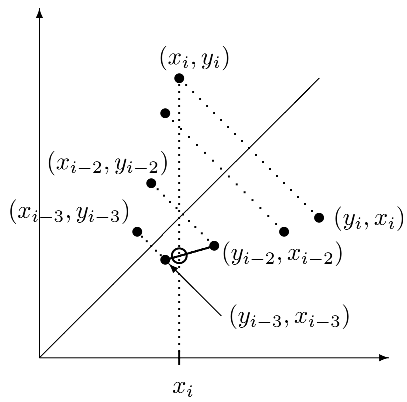

Fig. 3. Points on the function and on the inverse function with correspondences (slanted dotted lines). The value of the inverse function at support point x_i is, in this case, linearly interpolated from the values y_{i-3} and y_{i-2} so that $j = i - 3$.

function by

$$(y^{-1})'(y_j) = \frac{1}{y'(y^{-1}(y_j))} = \frac{1}{y'(x_j)}$$

$$= \frac{1}{F(y^{-1}(x_j), x_j)}$$
$$\left(y^{-1}\right)'(y_{j+1}) = \frac{1}{F(y^{-1}(x_{j+1}), x_{j+1})}.$$

A cubic polynomial $p_3(x)$ is thus computable from the four equations

$$p_3(y_j) = x_j,\ p_3(y_{j+1}) = x_{j+1},\ p_3'(y_j) = \frac{1}{F(y^{-1}(x_j), x_j)},$$

$$p_3'(y_{j+1}) = \frac{1}{F(y^{-1}(x_{j+1}), x_{j+1})}.$$

The inverse function value $y^{-1}(x_i)$ can now be interpolated as the value $p_3(x_i)$ for $x_i \in [y_j, y_{j+1}]$ if the inverse function values were known at x_j and x_{j+1}.

III. Algorithms

One difficulty of numerically solving ODEIs is that simple forward computations may fail. The reason is that the inverse solution function might be required to be known, at least approximately, for a larger argument than can be obtained by inverting the solution function up to that argument, see principle sketch in figure 4. Of course, extrapolation could be invoked but it performs poorly, in particular, close to the initial boundary.

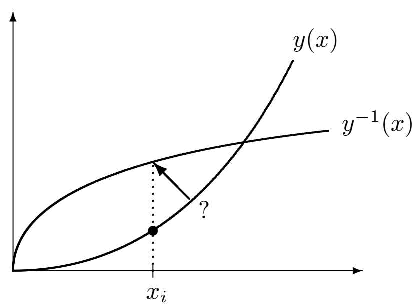

Fig. 4. Reasonably approximating the value of the inverse solution function at x_i, which would then computations allow to use the value $F(y^{-1}(x_i), x_i)$, requires to (approximately) know the function values at larger arguments.

A. Fixed point approach

In analogy to the explicit or forward Euler method for ordinary differential equations, see, for example [1], the derivative values of solution functions of an ODEI can be approximated by finite difference values as

$$\frac{y(x_i + h) - y(x_i)}{h} \approx y'(x_i) = F(y^{-1}(x_i), x_i).$$

This gives the approximating equations

$$y(x_{i+1}) = y(x_i) + h \cdot F(y^{-1}(x_i), x_i)$$

for $x_{i+1} = x_i + h$. In combination with an initial value $y(x_0) = y_0$ these equations, in turn, give rise to the vector-valued fixed point equation $y = \mathcal{F}(y)$ with

$$y = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} y_0 \\ \frac{1}{2} \cdot y_0 + \frac{1}{2} \cdot (y_1 + h \cdot F(g_1(y_0, \ldots, y_n), x_1)) \\ y_1 + h \cdot F(g_1(y_0, \ldots, y_n), x_1) \\ y_2 + h \cdot F(g_2(y_0, \ldots, y_n), x_2) \\ \vdots \\ y_{n-1} + h \cdot F(g_{n-1}(y_0, \ldots, y_n), x_{n-1}) \end{pmatrix}.$$

Functions $g_i(y_0, \ldots, y_n)$ indicate approximations of $y^{-1}(x_i)$. These can be obtained from procedure Test-and-invert which computes approximate inverses from the current linear approximation of the solution function. Since this will not work for $i = 0$, the computation of the second coordinate is exceptionally facilitated by linear interpolation between the first and the third coordinate.

Unlike the original Euler method, one step of these computations requires to know all current approximation values of a solution function rather than a single one. Consequently, an initial solution function is required instead of one initial boundary value only. Termination of the fixed point method may adhere to one of many distinct criteria. Here, it was chosen to halt computations when the squared differences between function values for two successive iterations summed over all support points differ less than a given threshold.

At best, convergence of the fixed point computations is slow and a sample result is sketched in figure 5. In this example, initialization with the golden section function decreased by the given initial value, which is the function $y(x) = (1/g)^{1/g} \cdot x^g - 200$, slightly speeds up the approximations as compared to linear initializations. Though the solution function is known to be convex prior to any computation, see above, initialization with a strictly concave function may be feasible. Yet, choices of initializing functions that have a different curvature than the solution function are delicate to select.

Two samples with trigonometric system functions leading to solutions that are neither convex nor concave are sketched in figure 6 with approximations based on 300 iterations. Solution functions are conjectured to be arithmetic quasiperiodic. A function $y(x)$ is understood to be arithmetic quasiperiodic if there are constants such that $y(x + a) = y(x) + b$ for all arguments x from the domain of the function [6].

B. Symmetric explicit method

Using whole functions in each iteration allows to symmetrize the explicit Euler method by averaging over approximations from above and below. The approximation from below of the forward Euler method for x_i instead of x_{i+1} is

$$y(x_i) = y(x_{i-1}) + h \cdot F(y^{-1}(x_{i-1}), x_{i-1})$$

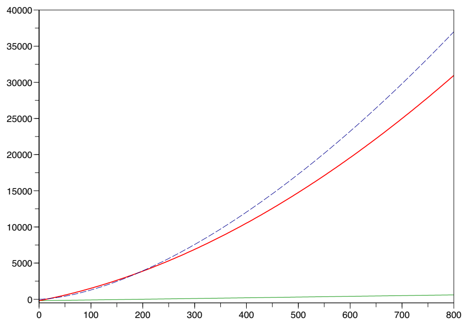

Fig. 5. Solution function (bold curve) over $Dom = [0, 800]$ computed by the fixed point method for $y'(x) = y^{-1}(x)$ with initial boundary value $y(0) = -200$. Initialization was obtained by the linear function $y(x) = x - 200$ (thin curve). Included for comparison is the golden section function $y(x) = (1/g)^{1/g} \cdot x^g$ solving the problem for the initial boundary value $y(0) = 0$ (dashed curve).

with positive increment value h. For the same increment value, the solution function can be approximated over support points from above by

$$\frac{y(x_{i+1} - h) - y(x_{i+1})}{-h} \approx y'(x_{i+1}) = F(y^{-1}(x_{i+1}), x_{i+1}).$$

This leads to the expressions

$$\begin{array}{rrcl} & y(x_{i+1} - h) & = & y(x_{i+1}) - h\, F(y^{-1}(x_{i+1}), x_{i+1}) \\ \Longrightarrow & y(x_i) & = & y(x_{i+1}) - h\, F(y^{-1}(x_{i+1}), x_{i+1}). \end{array}$$

Averaging the approximations over support points from above and below results in the formula

$$\begin{aligned} y(x_i) \;=\; & \frac{1}{2} \cdot \Big(y(x_{i-1}) + h \cdot F(y^{-1}(x_{i-1}), x_{i-1})\Big) \\ & + \frac{1}{2} \cdot \Big(y(x_{i+1}) - h\, F(y^{-1}(x_{i+1}), x_{i+1})\Big). \end{aligned}$$

Obviously, the leftmost and rightmost support point do not admit such an averaging approximation. As it may be difficult to compute the inverses at these points as well, it may be necessary to also exclude the second left and second right support point. In case of an initial value at the lower bound, only one support point at the lower bound and two support points at the upper bound must be excluded. The resulting vector-valued fixed point equation becomes

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_{n-2} \\ y_{n-1} \\ y_n \end{pmatrix} = \begin{pmatrix} y_0 \\ \frac{1}{2} \cdot y_0 + \frac{1}{2} \cdot y_2 \\ \frac{1}{2} \cdot y_1 + \frac{1}{2} \cdot y_3 \\ \vdots \\ \frac{1}{2} \cdot y_{n-3} + \frac{1}{2} \cdot y_{n-1} \\ y_{n-2} \\ y_{n-1} \end{pmatrix}$$

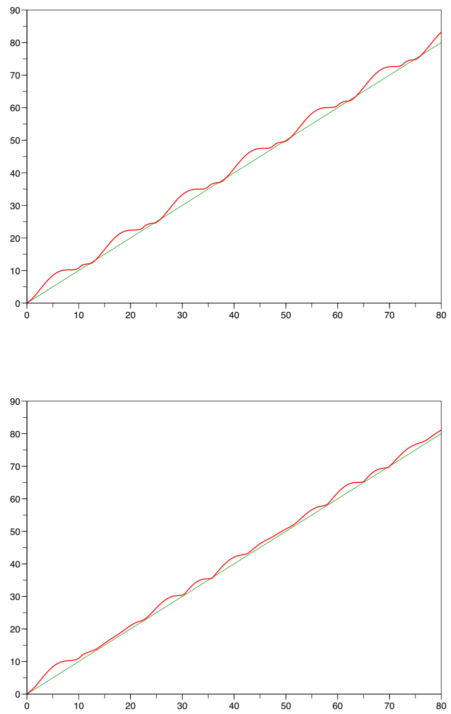

Fig. 6. Solutions (bold curves) over $Dom = [0, 80]$ computed by the fixed point method for $y'(x) = sin(y^{-1}(x)) + 1$ (top) and $y'(x) = sin(y^{-1}(x)) \cdot cos(0.1 \cdot y^{-1}(x)) + 1$ (bottom) both with initial boundary value $y(0) = 0$ and initialization function $y(x) = x$ (thin curves).

$$+h \cdot \begin{pmatrix} 0 \\ -\frac{1}{2}\, F(g_1(y_0, \ldots, y_n), x_2) \\ \frac{1}{2}\, F(g_1(y_0, \ldots, y_n), x_1) \\ \vdots \\ \frac{1}{2}\, F(g_{n-3}(y_0, \ldots, y_n), x_{n-3}) \\ F(g_{n-2}(y_0, \ldots, y_n), x_{n-2}) \\ F(g_{n-1}(y_0, \ldots, y_n), x_{n-1}) \end{pmatrix} - h \cdot \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2}\, F(g_1(y_0, \ldots, y_n), x_3) \\ \vdots \\ \frac{1}{2}\, F(g_{n-1}(y_0, \ldots, y_n), x_{n-1}) \\ 0 \\ 0 \end{pmatrix}$$

C. *Forward method*

A forward method of the Eulerian type or another is feasible in the special case of the inverse at any given support point being computable either from initial conditions or from values of the solution function to the left of that support point. Thus, a situation like in figure 4 must be avoided.

A sufficient condition therefore is that an increasing solution function satisfies $y(x_0) = x_0$ and $y(x) \geq x$ for all arguments x from the approximation domain. This condition implies

$y^{-1}(x_0) = x_0 = y(x_0)$ and $y^{-1}(x) \leq x \leq y(x)$. Informally, the solution function initially touches its inverse and lies above or touches its inverse over the whole approximating domain. This is equivalent to lie on or above the diagonal and exactly on the diagonal at the initial fixed point.

Under certain circumstances, forward methods can be modified to approximate sections of the solution function below the diagonal. This requires some fixed point to the right of such a section and then proceed backwards from the fixed point.

1) Initial value on the diagonal: As opposed to fixed point computations, no initial approximate solution function is required and approximations of the inverse function may or may not be interleaved with approximations of solution function. The condition $y_0 = x_0$ suffices for initialization resulting in two iteration schemes for the solution function and its inverse. They are denoted as discretized ODEI and discretized concatenated ODE.

$$\begin{aligned} y(x_{i+1}) &= y(x_i) + h\, F(y^{-1}(x_i), x_i), \quad i = 0, \ldots, n-1 \\ y^{-1}(x_i) &= y^{-1}(x_{i-1}) \\ &\quad + h\, \frac{1}{F(y^{-1}(y^{-1}(x_{i-1})), y^{-1}(x_{i-1}))},\ i = 1, \ldots, n-1. \end{aligned}$$

Both formulas are derived from the original ODEI and its reformulation with concatenation, see section 2, having the respective linear expansions $y(x_{i+1}) = y(x_i) + h \cdot y'(x_i)$ and $y^{-1}(x_i) = y^{-1}(x_{i-1}) + h \cdot (y^{-1})'(x_{i-1})$. Then the ODEI is applied to both expansion.

Individual iterations based on function and inverse

Making use of only the discretized ODEI and the numerical inversion procedure Test-and-invert results in the comparatively simple iterations

$$\begin{aligned} y_0 &= x_0 \\ y_1 &= y_0 + h\, F(y_0, x_0) \\ y_{i+1} &= y_i + h\, F(g_i(y_0, \ldots, y_i), x_i),\ i = 1, \ldots, n-1. \end{aligned}$$

The present conditions ensure that the functions g_i do only depend on the support points and on function values computed in previous iterations allowing the notation $g_i(y_0, \ldots, y_i)$. Since it avoids iterations over whole functions, the forward method is computationally less expensive than the fixed point method. In particular, the search for adjacent function values that enclose the current support point – as computed in step 1 of Test-and-invert – can be organized as a single sweep. This means that when an enclosing pair has been found, the enclosing pair of the next support point needs only be searched to the right.

Also, the forward method is robust in the sense that the solution function may slightly underpass the diagonal function when it has surpassed it to the left. For example, the solution from figure 6 (left) can be obtained by the forward method and that method uses far less than 1% of the computing time of the fixed point method. This speed-up adds to the comfort of not having to specify an initial solution function.

Individual iterations based on the inverse alone

Making use of only the concatenated ODE allows for an approximation of the inverse solution function and, after these computations have been completed, a subsequent inversion. The inverse function values $v_i \approx y^{-1}(x_i)$ are iteratively computable as

$$\begin{aligned} v_0 &= x_0 \\ v_1 &= v_0 + \frac{h}{F(v_0, x_0)} \\ v_{i+1} &= v_i + \frac{h}{F(w_i, v_i)},\ i = 1, \ldots, n-1. \end{aligned}$$

Once inverse values $y^{-1}(x_i)$ are known approximately, the second application of the inverse can be computed approximately by (1) a search for adjacent support points followed by (2) linear interpolation similar to single inversion computations. The steps for $w_i \approx y^{-1}(y^{-1}(x_i))$ are summarized in the next procedure.

Test-and-concatenate

1) Find $j \in \{0, 1, \ldots, n-1\}$ with $x_j < y^{-1}(x_i) \leq x_{j+1}$.
2) Compute $y^{-1}(y^{-1}(x_i)) = y^{-1}(x_j) + \frac{y^{-1}(x_i) - x_j}{x_{j+1} - x_j} \cdot (y^{-1}(x_{j+1}) - y^{-1}(x_j))$.

The index search from step 1 can be restricted to $j \in \{0, 1, \ldots, i-1\}$ which makes the scheme a forward computation.

Conjoint iterations

Individual iterations can be interleaved to form conjoint iterations of the solution function and its inverse. Out of the possible mixing patterns one with alternating value computations will be sketched. Alternating values will be used if the most recent value of the inverse computations affects the current function computation and vice versa. Values that reach to far back or too far ahead are discarded for interleaving in order to keep the computational control simple.

Forward-conjoint

1) Input $x_0 < x_1 < \ldots < x_n$ and system function $F(\cdot, \cdot)$.
   Initialization $y(x_0) = y^{-1}(x_0) = x_0$.
2) For $i = 0, \ldots, n-1$ do
   a) If $y(x_i) \in (x_i, x_{i+1})$ then compute
   $$y^{-1}(x_{i+1}) = x_i + (x_{i+1} - y(x_i)) \frac{1}{F(y^{-1}(x_i), x_i)};$$
   else compute
   $$\begin{aligned} y^{-1}(x_{i+1}) &= y^{-1}(x_i) \\ &\quad + h\, \frac{1}{F(y^{-1}(y^{-1}(x_i)), y^{-1}(x_i))}. \end{aligned}$$
   b) If $y^{-1}(x_{i+1}) \in (x_i, x_{i+1})$ then compute
   $$\begin{aligned} y(x_{i+1} &= x_{i+1} + (x_{i+1} - y^{-1}(x_{i+1})) \cdot \\ &\quad F(y^{-1}(y^{-1}(x_{i+1})), y^{-1}(x_{i+1})); \end{aligned}$$
   else compute
   $$y(x_{i+1}) = y(x_i) + h\, F(y^{-1}(x_i), x_i).$$

International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:5, No:12, 2011

3) Output $y(x_1), \ldots, y(x_n)$.

Inverse values and inverse values of inverse values are computable by the procedures Test-and-invert and Test-and-invert 2. An example showing gain in accuracy of the conjoint forward iterations over individual forward iterations is sketched in figure 7.

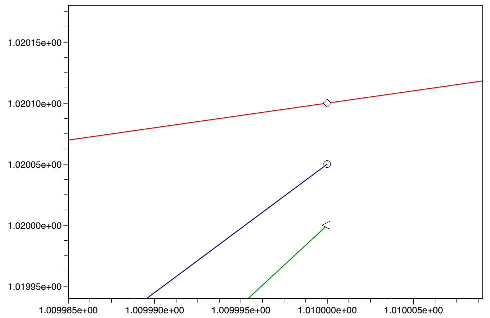

Fig. 7. Exact solution function $y(x) = x^2$ (top curve) of $y' = 2\cdot(y^{-1})^2$ in a small interval around $x_1 = 1.01$ with approximations starting at $x_0 = y_0 = 1$ for the conjoint forward iterations (middle curve) and individual forward computations based on the function and its inverse (bottom curve).

2) Given initial value off the diagonal: When the initial value lies above the diagonal, values of the inverse solution function are not coupled by the ODEI to values of the solution function for small arguments. Even initial boundary values for the solution function do not suffice so that additional information is required. This information will come as an initial boundary value of the inverse $y^{-1}(x_0) = z_0$ which is either provided explicitly or approximated by a slope condition. Both options are sketched beginning with the explicit additional provision, see figure 8.

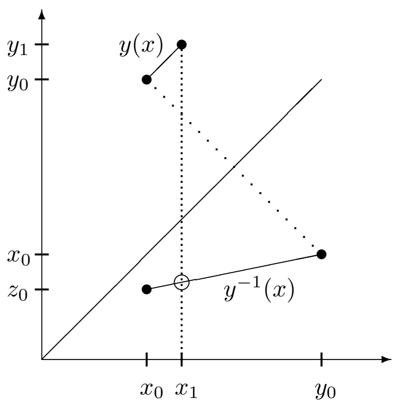

Fig. 8. Initial linear segments of the solution function and its inverse with extra specification of the initial value z_0 of the inverse function. This allows to approximate the slope of the solution function at x_1 because $y^{-1}(x_1)$ is now approximately given.

The initial boundary value of the inverse solution function is used as extra value allowing for linear interpolations that approximate the values $y^{-1}(x_1), y^{-1}(x_2)$ etc. until the argument exceeds the first proper value: $x_i \geq y_0$. Denoting these approximations by $g_i(z_0, y_0, \ldots, y_i)$ allows to specify the forward method for the initial boundary value $y(x_0) = y_0 \geq x_0$ as

$$\begin{aligned} y_1 &= y_0 + h\,F(y_0, x_0) \\ y_{i+1} &= y_i + h\,F(g_i(z_0, y_0, \ldots, y_i), x_i),\ i = 1, \ldots, n-1. \end{aligned}$$

Though the method is fast, errors may grow quite fast shown in figure 9. Improvements of the approximation can be achieved by overwriting the initial linear segment in each iteration in which the considered support point is still smaller than the initial function value y_0. The linear segment used for computing y_{i+1} would then linearly interpolate the inverse function over the domain $[x_{i-1}, y_i]$. However, the gain in accuracy from overwriting linear segments is moderate.

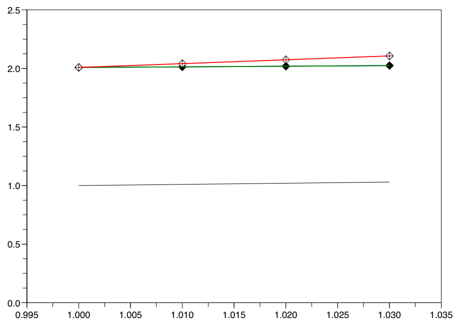

Fig. 9. Solution function $y(x) = (5/g)^{1/g} \cdot x^g$ (top curve) and forward approximation (middle curve) with exact boundary value $y_0 = y(1)$ and value $z_0 = 0.1$ for the ODEI $y' = 5 \cdot y^{-1}$ over the interval $[x_0, x_3] = [1, 1.03]$. Relative approximation errors at the support points x_1, x_2 and x_3 are 1.35%, 2.69% and 3.89% respectively. The error values are identical – up to three decimals – for the pure forward method and for the forward method with overwriting. Solutions lie above the diagonal (bottom curve).

3) Missing initial value off the diagonal: When an initial boundary value of the inverse function is not provided, a surrogate value can be chosen along the following argument. For ideal solution functions – rather than for linear approximations – the derivative of the inverse and the initial boundary value are related by

$$(y^{-1})'(y_0) = \frac{1}{y'(y^{-1}(y_0))} = \frac{1}{y'(x_0)}.$$

Evaluating the ODEI at the initial boundary value results in the exact equation $y'(x_0) = F(y^{-1}(x_0), x_0) = F(z_0, x_0)$. Replacing the derivative of the inverse by the slope of its linear segment yields the approximate equation $(y^{-1})'(y_0) = \frac{x_0 - z_0}{y_0 - x_0}$. Combining these expressions results in the equations

$$\begin{aligned} \frac{x_0 - z_0}{y_0 - x_0} &= \frac{1}{F(z_0, x_0)} \\ \Longrightarrow (x_0 - z_0)\,F(z_0, x_0) &= y_0 - x_0. \end{aligned}$$

Typically, the last equation cannot be solved for z_0 exactly but it may be approximately. Even more, a least squares approximation may be set up from the very beginning

$$z_0 = argmin_{z\in[0,x_0)} \Big((x_0 - z)\, F(z, x_0) - (y_0 - x_0)\Big)^2.$$

D. Methods based on numerical integration

In the sprit of Runge Kutta methods, solutions of ODEIs can be approximated by methods that make use of numerical integration. Therefore, the differential equation is rewritten as an integral equation and the integral is approximated by a Riemann sum. Different Riemann sums result in different overall computing schemes. In a simple version, the Riemann sum consists of two terms only and samples the function to be integrated at the left and right interval boundary.

$$\begin{aligned} y(x_{i+1}) &= y(x_i) + \Big(y(x_{i+1}) - y(x_i)\Big) \\ &= y(x_i) + \int_{x_i}^{x_{i+1}} y'(u)\, du \\ &\approx y(x_i) + \frac{h}{2}\, y'(x_i) + \frac{h}{2}\, y'(x_{i+1}) \\ &= y(x_i) + \frac{h}{2}\, F(y^{-1}(x_i), x_i) \\ &\quad + \frac{h}{2}\, F(y^{-1}(x_{i+1}), x_{i+1}). \end{aligned}$$

For the more frequently used Simpson rule, the approximation becomes

$$\begin{aligned} y(x_{i+1}) &\approx y(x_i) + \frac{h}{6}\, F\big(y^{-1}(x_i), x_i\big) \\ &\quad + \frac{4}{6}\, h\, F(y^{-1}(\frac{x_i + x_{i+1}}{2}), \frac{x_i + x_{i+1}}{2}) \\ &\quad + \frac{h}{6}\, F(y^{-1}(x_{i+1}), x_{i+1}). \end{aligned}$$

Since the inverse solution function is to be evaluated at several points in order to compute the "next" function value, the fixed point approach rather than the forward approach is the conceptually easier candidate to obtain a full computation scheme.

1) Fixed point method: The fixed point equation resulting from numerical integration with the foregoing approximation is similar to that for the symmetric explicit method. Again, the difficulty of computing inverse function values for the leftmost and rightmost support point result in different formulas involving these support points. For example, when using the numerical approximation that samples the integral without intermediate point, the fixed point iteration formula becomes

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_{n-1} \\ y_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_0 \\ y_1 \\ \vdots \\ y_{n-2} \\ y_{n-1} \end{pmatrix} + \begin{pmatrix} 0 \\ h\cdot F(g_1(y_0,\dots,y_n), x_1) \\ \frac{h}{2}\cdot F(g_1(y_0,\dots,y_n), x_1) \\ \vdots \\ \frac{h}{2}\cdot F(g_{n-2}(y_0,\dots,y_n), x_{n-2}) \\ h\cdot F(g_{n-1}(y_0,\dots,y_n), x_{n-1}) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \frac{h}{2}\cdot F(g_2(y_0,\dots,y_n), x_2) \\ \vdots \\ \frac{h}{2}\cdot F(g_{n-1}(y_0,\dots,y_n), x_{n-1}) \\ 0 \end{pmatrix}.$$

2) Forward method: Unlike in Runge Kutta methods for nere differential equations, present solution values in the argument of the system function need more intricate treatment than Taylor expansions; those would yield approximations with derivative values computable only by repeated evaluation of the system function; see, for example, [8]. For the simple case of the numerical integral approximation without intermediate point, the expansion here becomes

$$\begin{aligned} y(x_{i+1}) &\approx y(x_i) + \frac{h}{2}\, F(y^{-1}(x_i), x_i) \\ &\quad + \frac{h}{2}\, F(y^{-1}(x_{i+1}), x_{i+1}) \\ &\approx y(x_i) + \frac{h}{2}\, F(y^{-1}(x_i), x_i) + \frac{h}{2}\, F(y^{-1}(x_i) \\ &\quad + \frac{h}{F(y^{-1}(y^{-1}(x_i)), y^{-1}(x_i))}, x_{i+1}). \end{aligned}$$

The last approximate equation is derived similar to the forward computations for concatenated ODEs, see above. In case the solution function does not underpass the diagonal and when it satisfies an initial fixed point condition $y(x_0) = x_0$, the forward computation allow to solve the ODEI. The inverse function values are approximated by procedure Test-and-invert while their concatenations are approximated by procedure Test-and-concatenate. For the Simpson rule and similar rules, the forward computations need no more evaluation of inverse functions and concatenated inverse function though the system function is evaluated at additional points. This can be seen from the next approximate expansion.

$$\begin{aligned} y(x_{i+1}) &\approx y(x_i) + \frac{h}{6}\, F\big(y^{-1}(x_i), x_i\big) \\ &\quad + \frac{4}{6}\, h\, F(y^{-1}(\frac{x_i + x_{i+1}}{2}), \frac{x_i + x_{i+1}}{2}) \\ &\quad + \frac{h}{6}\, F(y^{-1}(x_{i+1}), x_{i+1}) \\ &\approx y(x_i) + \frac{h}{6}\, F(y^{-1}(x_i), x_i) \\ &\quad + \frac{4}{6}\, h\, F(y^{-1}(x_i) + \frac{h/2}{F(y^{-1}(y^{-1}(x_i)), y^{-1}(x_i))}, \\ &\qquad \frac{x_i + x_{i+1}}{2}) \\ &\quad + \frac{h}{6}\, F(y^{-1}(x_i) + \frac{h}{F(y^{-1}(y^{-1}(x_i)), y^{-1}(x_i))}, \\ &\qquad x_{i+1}). \end{aligned}$$

E. Error analysis

Forward computations for ODEIs are consistent and convergent similarly to ODEs. However, since computations of function inverses or function concatenations are involved, their approximations must also be accounted for in the error analysis.

1) Consistency for concatenated ODEs: Forward computations for concatenated ODEs with Lipschitz-bounded system functions, if applicable, are shown to be consistent of order one for twice continuously differentiable solution functions. To simplify the notation, the concatenated ODE is formulated for mere functions instead of their inverses as $y'(x) = f(y(y(x)), y(x))$.

In analogy to mere ODEs, an approximation scheme of difference equations is here understood to be consistent over some bounded closed region U if $\lim_{h\to 0} \max_{x_k \in U} |\delta_h(x_k)| = 0$ and it is consistent of order p if $\max_{x_k \in U} |\delta_h(x_k)| = O(h^p)$ for the pointwise error

$$\delta_h(x_k) = \frac{y(x_{k+1}) - y(x_k)}{h} - f\big(y(x_j) + \frac{y(x_k) - x_j}{h} \cdot (y(x_{j+1}) - y(x_j)), y(x_k)\big).$$

For a linear approximation of concatenated function values, the index $j = j(k)$ is chosen such that $x_j \le y(x_k) \le x_{j+1}$. This linear approximation results from the exact solution inserted into the procedure Test-and-concatenate. The pointwise error results from inserting any increasing exact solution function of the concatenated ODE into the equations of the approximation scheme; no approximate solution is involved so that the error is a discretization error rather than an approximation error. Discretization refers to both, the differentiation – as for mere ODEs – and the concatenation – as is special here.

It can now be seen from a Taylor expansion with second order error term that

$$\begin{aligned}\delta_h(x_k) &= \frac{y(x_{k+1}) - y(x_k)}{h} - f(y(y(x_k)), y(x_k)) \\ &\quad + f(y(y(x_k)), y(x_k)) - f\big(y(x_j) \\ &\qquad + \frac{y(x_k) - x_j}{h} \cdot (y(x_{j+1}) - y(x_j)), y(x_k)\big) \\ &= y'(x_k) + \frac{h}{2}\, y''(x_k^*) - f(y(y(x_k)), y(x_k)) \\ &\quad + f(y(y(x_k)), y(x_k)) - f\big(y(x_j) \\ &\qquad + \frac{y(x_k) - x_j}{h} \cdot (y(x_{j+1}) - y(x_j)), y(x_k)\big) \\ &= \frac{h}{2}\, y''(x_k^*) + f(y(y(x_k)), y(x_k)) - f\big(y(x_j) \\ &\qquad + \frac{y(x_k) - x_j}{h} \cdot (y(x_{j+1}) - y(x_j)), y(x_k)\big)\end{aligned}$$

with $x_k^* \in (x_k, x_{k+1})$. The last equality follows from function $y(x)$ being a solution of the concatenated ODE. in contrast to mere ODEs, the resulting expression for the discretization error still contains the system function and, thus, complicates the analysis. The solution function of the concatenated ODE being continuous over U and the system function being Lipschitz-bounded there leads to

$$\begin{aligned}|\delta_h(x_k)| &\le \frac{h}{2}\, |y''(x_k^*)| + L \cdot |y(y(x_k)) - y(x_j) \\ &\qquad - \frac{y(x_k) - x_j}{h} \cdot (y(x_{j+1}) - y(x_j))| \\ &\le h \cdot M.\end{aligned}$$

Lipschitz-boundedness of the solution function follows from it being (twice) continuously differentiable over a bounded closed region. With Lipschitz constant L_y and $z = y(x_k)$ this allows to bound the last term which then completes the argument of the consistency result:

$$\begin{aligned}&|y(x_j) + \frac{z - x_j}{h} \cdot (y(x_{j+1}) - y(x_j)) - y(z)| \\ &= |\frac{x_{j+1} - z}{x_{j+1} - x_j}\, (y(x_j) - y(z)) \\ &\qquad + \frac{z - x_j}{x_{j+1} - x_j}\, (y(x_{j+1}) - y(z))| \\ &\le \frac{x_{j+1} - z}{x_{j+1} - x_j}\, |y(x_j) - y(z)| \\ &\qquad + \frac{z - x_j}{x_{j+1} - x_j}\, |y(x_{j+1}) - y(z))| \\ &\le \frac{x_{j+1} - z}{x_{j+1} - x_j}\, L_y\, |x_j - z| + \frac{z - x_j}{x_{j+1} - x_j}\, L_y\, |x_{j+1} - z| \\ &\le h\, L_y.\end{aligned}$$

2) Consistency for ODEIs: An approximation scheme for the ODEIs based on difference equations is understood to be consistent (of order p) over some bounded closed region U if the corresponding limit condition holds as for ODEs with concatenation with modified pointwise error term

$$\delta_h(x_k) = \frac{y(x_{k+1}) - y(x_k)}{h} - F\big(x_j + \frac{x_k - y(x_j)}{y(x_{j+1}) - y(x_j)} \cdot (x_{j+1} - x_j), x_k\big).$$

The index $j = j(k)$ is chosen to approximate the inverse function at x_k via $y(x_j) \le x_k \le y(x_{j+1})$. For twice differentiable solution functions and a Lipschitz bounded system function the forward computations of the ODEI are consistent or order one. The argument is similar to those for concatenated ODEs, namely by a Taylor expansion for the error term resulting in

$$\begin{aligned}\delta_h(x_k) &= \frac{h}{2}\, y''(x_k^*) + F(y^{-1}(x_k), x_k) \\ &\quad - F\big(x_j + \frac{x_k - y(x_j)}{y(x_{j+1}) - y(x_j)} \cdot (x_{j+1} - x_j), x_k\big)\end{aligned}$$

for $x_k^* \in (x_k, x_{k+1})$. The Lipschitz bound of the system function allows to reduce the remaining considerations to the first arguments. With $\alpha = \frac{y(x_{j+1}) - x_k}{y(x_{j+1}) - y(x_j)} \in (0, 1)$ the difference between these arguments can be bounded as

$$\begin{aligned}
& |x_j + \frac{x_k - y(x_j)}{y(x_{j+1}) - y(x_j)} \cdot (x_{j+1} - x_j) - y^{-1}(x_k)| \\
= \;& |\alpha \cdot (x_j - y^{-1}(x_k)) + (1-\alpha) \cdot (x_{j+1} - y^{-1}(x_k))| \\
\leq \;& \alpha \cdot |y^{-1}(y(x_j)) - y^{-1}(x_k))| \\
& +(1-\alpha) \cdot |y^{-1}(y(x_{j+1})) - y^{-1}(x_k))| \\
\leq \;& \alpha \cdot L_y^- \cdot |y(x_j) - x_k| + (1-\alpha) \cdot L_y^- \cdot |y(x_{j+1}) - x_k| \\
\leq \;& \alpha \cdot L_y^- \cdot |y(x_j) - y(x_{j+1})| \\
& +(1-\alpha) \cdot L_y^- \cdot |y(x_{j+1}) - y(x_j)| \\
\leq \;& \alpha \cdot L_y^- \cdot L_y\, h + (1-\alpha) \cdot L_y^- \cdot L_y\, h \\
= \;& L_y^- \cdot L_y\, h.
\end{aligned}$$

Thus, the discretization error satisfies the linear bound $|\delta_h(x_k)| \leq \frac{h}{2}\ |y''(x_k^*)| + L_y^- \cdot L_y\, h = O(h)$ on the closed bounded region.

3) Convergence: An approximation scheme for an ODEI is convergent over some bounded closed region U if $\lim_{h\to 0}$ $\max_{x_k \in U}$ $|\Delta_h(x_k)| = 0$ and it is convergent of order p if $\max_{x_k\in U} |\Delta_h(x_k)| = O(h^p)$ for the approximation error $\Delta_h(x_k) = y_k - y(x_k)$; y_k and $y(x_k)$ denote values of an approximate function and an exact solution function, respectively. As in mere ODEs, the approximation error for ODEIs will be bounded with the help of the discretization error.

A featuring difficulty of establishing an order of convergence of the forward iterations is that, simultaneously, convergence of that order of the approximate inverse towards the inverse solution function must be established. A sufficient condition therefore is that the forward computations are accurate enough to approximate the inverse for all required arguments in the correct interval of successive support points. This means that

$$y(x_j) \leq x_k \leq y(x_{j+1}) \Longrightarrow y_j \leq x_k \leq y_{j+1}.$$

Under this "accuracy" condition, Lipschitz boundedness of the system function, double differentiability of the solution function and the initial fixed point condition $y(x_0) = x_0$ imply that the forward computations is convergent of order one on a suitable bounded closed interval.

IV. Bounds and closed-form approximations

Bounds for system functions can imply bounds for solution functions. Such bounds will be considered for the special case of initial boundary valuw zero at the origin. Let the system function have non-negative values and a positive upper bound so that $0 \leq F(a,b) \leq M$ for all suitable arguments. Then $y'(x) \leq M$ so that $y(x) \leq M\,x$. This upper bound on the solution implies a lower bound on is inverse $y^{-1}(x) \geq 1/M\,x$ whenever $M > 1$. Inserting the lower bound or a steeper linear function into the system functions allows to define a function via its derivative: $y'_{lin}(x) = F(\alpha\, x, x)$. Indefinite integration yields the linearly induced approximating function

$$y_{lin}(x) = \int_0^x F(\alpha\, u, u)\, du.$$

The approximation satisfies the initial boundary value $y_{lin}(x) = 0$. A sample situation with $F(a,b) = sin(a) + 1$ and upper bound $M = 2$ is depicted in figure 10. Linearly induced approximating function can be found, similarly, when the system function has a strictly positive lower bound. Empirical investigations suggest that solutions of the ODEI $y'(x) = sin(y^{-1}(x)) + A$, with $y(0) = 0$ and $A \geq 1$, have periodic derivatives with period of about $(1 + A)\, 2\,\pi$. Every such solution satisfies

$$(A-1)\,x \leq y(x) = \int_0^x sin(y^{-1}(u))\,du + A\,x \leq (A+1)\,x.$$

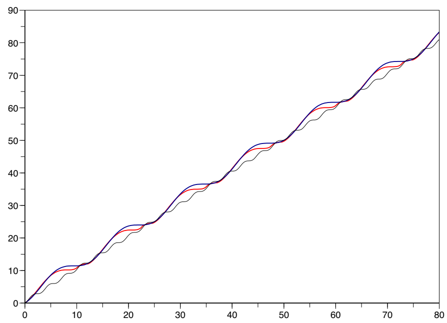

Fig. 10. Solution (bold curve, partially occluded) over $Dom = [0, 80]$ for $y'(x) = sin(y^{-1}(x)) + 1$ with initial boundary value $y(0) = 0$ as in figure 6 (left). Linearly induced approximating functions $y_{lin}(x) = x - 2 \cdot \cos(0.5\,x) + 2$ for $\alpha = 0.5$ (bold curve, same frequency and "mostly" above the solution) and $y_{lin}(x) = x - 0.5 \cdot \cos(2\,x) - 0.5$ for $\alpha = 2$ (thin, high frequency curve).

V. Conclusion

First order differential equations with inverted functions have been introduced and their numerical solvability has been shown to be feasible. Accuracy is affected by approximations of the inverse to at least the same degree as by the proper function approximation. This can be expected, also, for more complex differential equations of the given type.

References

[1] Birkhoff, G., Rota, G.-C., "Ordinary differential equations", 2nd ed., Wiley, New York, 1969.
[2] Gilsinn, D.E., Potra, F.A., "Integral operators and delay differential equations", Journal of Integral Equations and Applications 18, 2006, p. 297-336.
[3] Huylebrouck, D., "Golden Gray", Forma 19, 2004, p. 111-120.
[4] Kämpke, T., "On duality transformations between Lorenz curves and distribution functions", FAW/n, Ulm, 2011, www.faw-neu-ulm.de/artikel.
[5] Koepf, W., "Taylor polynomials of implicit functions, of inverse functions, and of solutions of ordinary differential equations", Complex Variables 25, 1994, p. 23-33.
[6] PlanetMath, "Quasiperiodic functions", planetmath.org/encyclopedia/QuasiperiodicFunction.html, accessed June 30, 2011.
[7] Scilab, "The free platform for numerical computation", www.scilab.org.
[8] Stoer, J., Bulirsch, R., "Numerische Mathematik 2", 5th ed., Springer, Berlin, 2005.
[9] Smart, D.R., "Fixed point theorems", 2nd printing, Cambridge University Press, Cambridge, 1980.
[10] Schnell, S., Mendoza, C., "A formula for integrating inverse functions", The Mathematical Gazette 84, 2000, p. 103-104.
[11] Zill, D.G., Wright, W.S, "Advanced engineering mathematics", 4th ed., Jones & Bartlett Learning, Burlington, 2010.