Some Recent Advances in Validated Methods for IVPs for ODEs [1]

Kenneth R. Jackson [a] and Nedialko S. Nedialkov [b]

[a] *Department of Computer Science, University of Toronto, Toronto, Ontario, M5S 3G4, Canada. E-mail: krj@cs.toronto.edu*

[b] *Department of Computing and Software, McMaster University, Hamilton, Ontario, L8S 4L7, Canada. E-mail: nedialk@mcmaster.ca*

Abstract

Compared to standard numerical methods for initial value problems (IVPs) for ordinary differential equations (ODEs), validated methods (often called interval methods) for IVPs for ODEs have two important advantages: if they return a solution to a problem, then (1) the problem is guaranteed to have a unique solution, and (2) an enclosure of the true solution is produced.

We present a brief overview of interval Taylor series (ITS) methods for IVPs for ODEs and discuss some recent advances in the theory of validated methods for IVPs for ODEs. In particular, we discuss an interval Hermite-Obreschkoff (IHO) scheme for computing rigorous bounds on the solution of an IVP for an ODE, the stability of ITS and IHO methods, and a new perspective on the wrapping effect, where we interpret the problem of reducing the wrapping effect as one of finding a more stable scheme for advancing the solution.

Key words: validated methods, interval methods, initial value problems, ordinary differential equations, Taylor series, stability, wrapping effect, simultaneous iteration, QR algorithm

---

1 The Initial Value Problem

We consider the set of autonomous IVPs

[1] This work was supported in part by the Natural Sciences and Engineering Research Council of Canada, the Information Technology Research Centre of Ontario, and Communications and Information Technology Ontario.

$$y'(t) = f(y) \tag{1}$$
$$y(t_0) \in [y_0], \tag{2}$$

where $t \in [t_0, t_m]$ for some $t_m > t_0$. Here t_0 and $t_m \in \mathbb{R}$, $f \in C^{k-1}(D)$, $k \geq 2$ (k is the order of the truncation error in the methods that we present later), $D \subseteq \mathbb{R}^n$ is open, $f : D \to \mathbb{R}^n$, and $[y_0] \subseteq D$. The condition (2) permits the initial value $y(t_0)$ to be in an interval, rather than specifying a particular value. We assume that the representation of f contains a finite number of constants, variables, elementary operations, and standard functions. Since we assume $f \in C^{k-1}(D)$, we exclude functions that contain, for example, branches, abs, or min. For expositional convenience, we consider only autonomous systems. This is not a restriction of consequence, since a nonautonomous system of ODEs can be converted into an autonomous one. Moreover, the methods discussed here can be extended easily to nonautonomous systems.

We consider a grid $t_0 < t_1 < \cdots < t_m$, which is not necessarily equally spaced, and denote the stepsize from t_{j-1} to t_j by $h_{j-1} = t_j - t_{j-1}$. The step from t_{j-1} to t_j is referred to as the jth step. We denote the solution of (1) with an initial condition $y(t_{j-1}) = y_{j-1}$ at t_{j-1} by $y(t; t_{j-1}, y_{j-1})$. For an interval, or an interval vector $[y_{j-1}]$, we denote by $y(t; t_{j-1}, [y_{j-1}])$ the set of solutions

$$\{\, y(t; t_{j-1}, y_{j-1}) \mid y_{j-1} \in [y_{j-1}] \,\}.$$

Our goal is to compute interval vectors $[y_j]$, $j = 1, 2, \ldots, m$, that are guaranteed to contain the solution of (1–2) at $t_1, t_2, \ldots, t_m$. That is,

$$y(t_j; t_0, [y_0]) \subseteq [y_j], \quad \text{for } j = 1, 2, \ldots, m.$$

Standard numerical methods for IVPs for ODEs attempt to compute an approximate solution that satisfies a user-specified tolerance. These methods are usually robust and reliable for most applications, but it is possible to find examples for which they return inaccurate results. On the other hand, if a validated method for IVPs for ODEs returns successfully, it not only produces a guaranteed bound for the true solution, but also verifies that a unique solution to the problem $y'(t) = f(y)$, $y(t_0) = y_0$ exists for all $y_0 \in [y_0]$.

There are situations when guaranteed bounds are desired or needed. For example, a guaranteed bound on the solution could be used to prove a theorem [28]. Also, some calculations may be critical to the safety or reliability of a system. Therefore, it may be necessary or desirable to ensure that the true solution is within the computed bounds.

One reason validated solutions to IVPs for ODEs have not been popular in the past is that their computation typically requires considerably more time and memory than does that of standard methods. However, now that "chips are cheap", it seems natural to shift the burden of determining the reliability of a numerical solution from the user to the computer — at least for standard problems that do not require a very large amount of computer resources.

In addition, there are situations where interval methods for IVPs for ODEs may not be computationally more expensive than standard methods. For example, many ODEs arising in practical applications contain parameters. Often these parameters cannot be measured exactly, but are known to lie in certain intervals, as, for example, in economic models or in control problems. In these situations, a user might want to compute solutions for ranges of parameters. If a standard numerical method is used, it has to be executed many times with different parameters, while an interval method can "capture" all the solutions at essentially no extra cost.

Important developments in the area of validated solutions of IVPs for ODEs are the interval methods of Moore [19], Krückeberg [13], Eijgenraam [8], and Lohner [17]. All these methods are based on Taylor series. One reason for the popularity of this approach is the simple form of the error term. In addition, the Taylor series coefficients can be readily generated by automatic differentiation, and both the order of the method and its stepsize can be changed easily from step to step.

Usually, validated methods for IVPs for ODEs are one-step methods, where each step consists of two phases:

Algorithm I: validate existence and uniqueness of the solution with some stepsize, and
Algorithm II: compute a tight enclosure for the solution.

The main difficulty in Algorithm I is how to validate existence and uniqueness of the solution with a stepsize that corresponds to the order of Algorithm II. The main obstacle in Algorithm II is how to reduce the so-called *wrapping effect*, which arises when the solution set, which is not generally a box, is enclosed in — or wrapped by — a box on each integration step, thus introducing overestimations in the computed bounds. Currently, Lohner's QR-factorization method is the most-effective standard scheme for reducing the wrapping effect.

The methods considered in this paper are based on high-order Taylor series expansions with respect to time. Recently, Berz and Makino [4] proposed a method for reducing the wrapping effect that employs high-order Taylor series expansions in both time and the initial conditions. Berz and Makino's scheme validates existence and uniqueness and also computes tight bounds on the solution at each grid point t_j in one phase, while the methods considered here separate these tasks into two phases, as noted above.

The purpose of this paper is to present a brief overview of interval Taylor series (ITS) methods for IVPs for ODEs and to discuss some recent advances in the area of validated ODE solving. In particular, we discuss

- an interval Hermite-Obreschkoff (IHO) scheme for computing tight enclosures on the solution [20, 21];
- instability in interval methods for IVPs for ODEs due to the associated formula for the truncation error [20, 21], which appears to make it difficult to derive effective validated methods for stiff problems; and
- a new perspective on the wrapping effect [22], where we view the problem of reducing the wrapping effect as one of finding a more stable scheme for advancing the solution.

Section 2 introduces interval-arithmetic operations, enclosing ranges of functions, and automatic generation of Taylor series coefficients. In Section 3, we briefly discuss methods for validating existence and uniqueness of the solution, explain the wrapping effect, and describe Lohner's QR-factorization method for reducing the wrapping effect.

In Section 4, we outline the IHO method. Section 5 shows that the stability of the ITS and the IHO methods depends not only on the formula for advancing the solution, as in point methods for IVPs for ODEs, but also on the associated formula for the truncation error. Section 6 explains how the wrapping effect can be viewed as a source of instability in interval methods for IVPs for ODEs and how Lohner's QR-factorization scheme improves the stability of an interval method.

2 Interval Arithmetic

The set of intervals on the real line $\mathbb{R}$ is defined by

$$\mathbb{IR} = \{\, [a] = [\underline{a}, \bar{a}] \mid \underline{a}, \bar{a} \in \mathbb{R},\ \ \underline{a} \le \bar{a} \,\}\,.$$

If $\underline{a} = \bar{a}$ then $[a]$ is a *point* interval; if $\underline{a} \ge 0$ then $[a]$ is *nonnegative* ($[a] \ge 0$); and if $\underline{a} = -\bar{a}$ then $[a]$ is *symmetric*. Two intervals $[a]$ and $[b]$ are equal if $\underline{a} = \underline{b}$ and $\bar{a} = \bar{b}$.

Let $[a]$ and $[b] \in \mathbb{IR}$, and $\circ \in \{+, -, *, /\}$. The interval-arithmetic operations are defined [19] by

$$[a] \circ [b] = \{\, x \circ y \mid x \in [a]\,,\ y \in [b] \,\}\,, \quad 0 \notin [b] \ \text{ when } \circ = /, \tag{3}$$

which can be written in the following equivalent form (we omit $*$ in the notation):

$$[a]+[b]=\left[\underline{a}+\underline{b},\bar{a}+\bar{b}\right], \tag{4}$$
$$[a]-[b]=\left[\underline{a}-\bar{b},\bar{a}-\underline{b}\right], \tag{5}$$
$$[a]\,[b]=\left[\min\left\{\underline{a}\underline{b},\underline{a}\bar{b},\bar{a}\underline{b},\bar{a}\bar{b}\right\},\ \max\left\{\underline{a}\underline{b},\underline{a}\bar{b},\bar{a}\underline{b},\bar{a}\bar{b}\right\}\right], \tag{6}$$
$$[a]\,/\,[b]=[\underline{a},\bar{a}]\left[1/\bar{b},1/\underline{b}\right],\quad 0\notin[b]\,. \tag{7}$$

The strength of interval arithmetic when implemented on a computer is in computing rigorous enclosures of real operations by including rounding errors in the computed bounds. To include such errors, we round the real intervals in (4–7) outwards, thus obtaining machine interval arithmetic. For example, when adding intervals, we round $\underline{a}+\underline{b}$ down and round $\bar{a}+\bar{b}$ up. For simplicity of the discussion, we assume real interval arithmetic in this paper. Because of the outward roundings, intervals computed in machine interval arithmetic always contain the corresponding real intervals.

Definition (3) and formulas (4–6) can be extended to vectors and matrices. If the components of a vector or matrix are intervals, we have interval vectors and matrices, respectively. The arithmetic operations involving interval vectors and matrices are defined by the standard formulas, except that reals are replaced by intervals and real arithmetic is replaced by interval arithmetic in the associated computations.

We have an inclusion of intervals

$$[a]\subseteq[b]\iff\underline{a}\geq\underline{b}\text{ and }\bar{a}\leq\bar{b}.$$

The interval-arithmetic operations are *inclusion monotone*. That is, for real intervals $[a]$, $[a_1]$, $[b]$, and $[b_1]$ such that $[a]\subseteq[a_1]$ and $[b]\subseteq[b_1]$,

$$[a]\circ[b]\subseteq[a_1]\circ[b_1]\,,\quad\circ\in\{+,-,*,/\}\,. \tag{8}$$

Although interval addition and multiplication are associative, the distributive law does not hold in general [1]. That is, we can easily find three intervals $[a]$, $[b]$, and $[c]$, for which

$$[a]([b]+[c])\neq[a][b]+[a][c].$$

However, for any three intervals $[a]$, $[b]$, and $[c]$, the *subdistributive law*

$$[a]\,([b]+[c]) \subseteq [a]\,[b]+[a]\,[c]\,,$$

does hold. Moreover, there are important cases in which the distributive law

$$[a]\,([b]+[c]) = [a]\,[b]+[a]\,[c]$$

does hold. For example, it holds if $[b]\,[c] \geq 0$, if $[a]$ is a point interval, or if $[b]$ and $[c]$ are symmetric. In particular, for $\alpha \in \mathbb{R}$, which can be interpreted as the point interval $[\alpha, \alpha]$, and intervals $[b]$ and $[c]$, we have

$$\alpha([b]+[c]) = \alpha[b]+\alpha[c].$$

For an interval $[a]$, we define width and midpoint of $[a]$ as

$$\mathrm{w}([a]) = \bar{a} - \underline{a} \quad \text{and} \tag{9}$$

$$\mathrm{m}([a]) = (\bar{a} + \underline{a})/2, \tag{10}$$

respectively [19]. Width and midpoint are defined component-wise for interval vectors and matrices.

Using (4–6) and (9), one can easily show that

$$\mathrm{w}([a] \pm [b]) = \mathrm{w}([a]) + \mathrm{w}([b]) \quad \text{and} \tag{11}$$

$$\mathrm{w}(\alpha[a]) = |\alpha|\,\mathrm{w}([a]), \tag{12}$$

for any $[a]$, $[b] \in \mathbb{IR}$ and $\alpha \in \mathbb{R}$.

The equalities (11) and (12) also hold when $[a]$ and $[b]$ are interval vectors. If A is an $n \times n$ real matrix and $[a]$ is an n dimensional interval vector, then from (11–12),

$$\mathrm{w}(A[b]) = |A|\mathrm{w}([b]), \tag{13}$$

where $|A|$ is obtained by taking absolute values on each component of A.

2.1 Ranges of Functions

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a function on $D \subseteq \mathbb{R}^n$. The interval-arithmetic evaluation of f on $[a] \subseteq D$, which we denote by $f([a])$, is obtained by replacing each occurrence of a real variable with a corresponding interval, by replacing the standard functions with enclosures of their ranges, and by performing interval-arithmetic operations instead of the real operations. From (8), the range of f, $\{ f(x) \mid x \in [a] \}$, is always contained in $f([a])$. Although $f([a])$ is not unique, since expressions that are mathematically equivalent for scalars, such as $x(y+z)$ and $xy + xz$, may have different values if x, y, and z are intervals, a value for $f([a])$ can be determined from the code list, or computational graph, for f. This implies that, if we rearrange an interval expression, we may obtain tighter bounds.

If $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable on $D \subseteq \mathbb{R}^n$ and $[a] \subseteq D$, then, for any y and $b \in [a]$, $f(y) = f(b) + f'(\eta)(y - b)$ for some $\eta \in [a]$ by the mean-value theorem. Thus,

$$f(y) \in f_M([a], b) \equiv f(b) + f'([a])([a] - b) \tag{14}$$

[19]. The mean-value form, $f_M([a], b)$, is popular in interval methods since it often gives tighter enclosures for the range of f than the straightforward interval-arithmetic evaluation of f itself. Like f, f_M is not uniquely defined, but a value for it can be determined from the code lists for f and f'.

2.2 Automatic Generation of Taylor Coefficients

Since the interval methods considered here use Taylor series coefficients, we outline the scheme for their generation. More details can be found in [3] or [19], for example.

If we know the Taylor coefficients $(u)_i \equiv u^{(i)}/i!$ and $(v)_i \equiv v^{(i)}/i!$ for $i = 0, 1, \ldots, p$ for two functions u and v, then we can compute the pth Taylor coefficient of $u \pm v$, uv, and u/v by standard calculus formulas [19].

We introduce the sequence of functions

$$\begin{aligned} f^{[0]}(y) &= y, \\ f^{[i]}(y) &= \frac{1}{i}\left(\frac{\partial f^{[i-1]}}{\partial y} f\right)(y), \quad \text{for } i \geq 1. \end{aligned}$$

For the IVP $y' = f(y)$, $y(t_j) = y_j$, the ith Taylor coefficient of $y(t; t_j, y_j)$ at t_j, $(y_j)_i \equiv y^{(i)}(t_j)/i!$, satisfies

$$(y_j)_i = f^{[i]}(y_j) = \frac{1}{i}\big(f(y_j)\big)_{i-1}, \quad i \geq 1, \tag{15}$$

where $\left(f(y_j)\right)_{i-1}$ is the $(i-1)$st Taylor coefficient of f evaluated at y_j. That is,

$$\left(f(y_j)\right)_{i-1} = \frac{1}{(i-1)!} \left. \frac{d^{i-1} f\left(y(t;t_j,y_j)\right)}{dt^{i-1}} \right|_{y(t)=y_j} .$$

Using (15) and formulas for the Taylor coefficients of sums, products, quotients, and the standard functions, we can recursively evaluate $(y_j)_i$, for $i \geq 1$. It can be shown that, to generate k Taylor coefficients, we need at most $O(k^2)$ times as much computational work as we require to evaluate $f(y)$ [19]. Note that this is far more efficient than the standard symbolic generation of Taylor coefficients.

If we have a procedure to compute the point Taylor coefficients $f^{[i]}(y_j)$ of $y(t;t_j,y_j)$ and perform the computations in interval arithmetic with $[y_j]$ instead of y_j, to compute $f^{[i]}([y_j])$, we obtain a procedure to compute the interval Taylor coefficients of $y(t;t_j,y_j)$ at t_j, since

$$\left\{ \left. \left(y(t;t_j,y_j)\right)_i \right|_{t=t_j} \; : \; y_j \in [y_j] \right\} \subseteq f^{[i]}([y_j]).$$

3 Overview of Interval Methods for IVPs for ODEs

3.1 Algorithm I: Validating Existence and Uniqueness of the Solution

Using the Picard-Lindelöf operator and the Banach fixed-point theorem, one can show that if h_{j-1} and $[\tilde{y}^0_{j-1}] \supseteq [y_{j-1}]$ satisfy

$$[\tilde{y}_{j-1}] = [y_{j-1}] + [0, h_{j-1}]\, f\left([\tilde{y}^0_{j-1}]\right) \subseteq [\tilde{y}^0_{j-1}], \tag{16}$$

then (1) with $y(t_{j-1}) = y_{j-1}$ has a unique solution $y(t;t_{j-1},y_{j-1})$ that satisfies $y(t;t_{j-1},y_{j-1}) \in [\tilde{y}_{j-1}]$ for all $t \in [t_{j-1}, t_j]$ and all $y_{j-1} \in [y_{j-1}]$, [8].

We refer to a method based on (16) as a first-order enclosure method. Such a method can be easily implemented, but a serious disadvantage of this approach is that it often restricts the stepsize Algorithm II could take.

One can obtain methods that enable larger stepsizes by using polynomial enclosures [18] or more Taylor series terms in the sum in (16), thus obtaining a high-order Taylor series enclosure method [6, 24]. In the latter, we determine h_{j-1} and $[\tilde{y}_{j-1}]$ such that

$$\sum_{i=0}^{k-1}(t-t_{j-1})^i f^{[i]}(y_{j-1}) + (t-t_{j-1})^k f^{[k]}([\tilde{y}_{j-1}]) \subseteq [\tilde{y}_{j-1}] \tag{17}$$

holds for all $t \in [t_{j-1}, t_j]$ and all $y_{j-1} \in [y_{j-1}]$. Then (1) with $y(t_{j-1}) = y_{j-1}$ has a unique solution $y(t; t_{j-1}, y_{j-1})$ that satisfies $y(t; t_{j-1}, y_{j-1}) \in [\tilde{y}_{j-1}]$ for all $t \in [t_{j-1}, t_j]$ and all $y_{j-1} \in [y_{j-1}]$.

In [24], we show that, for many problems, an interval method based on (17) is more efficient than one based on (16).

3.2 Algorithm II: Computing a Tight Enclosure of the Solution

Consider the Taylor series expansion

$$y_j = y_{j-1} + \sum_{i=1}^{k-1} h_{j-1}^i f^{[i]}(y_{j-1}) + h_{j-1}^k f^{[k]}(y; t_{j-1}, t_j), \tag{18}$$

where $y_{j-1} \in [y_{j-1}]$, and $f^{[k]}(y; t_{j-1}, t_j)$ denotes $f^{[k]}$ with its lth component ($l = 1, 2, \ldots, n$) evaluated at $y(\xi_{j-1,l})$ for some $\xi_{j-1,l} \in [t_{j-1}, t_j]$. Using the a priori bounds $[\tilde{y}_{j-1}]$ on $y(t)$ for $t \in [t_{j-1}, t_j]$, we can enclose the local truncation error of the Taylor series (18) on $[t_{j-1}, t_j]$ by $h_{j-1}^k f^{[k]}([\tilde{y}_{j-1}])$ for any $y_{j-1} \in [y_{j-1}]$. We can also bound the ith Taylor coefficient $f^{[i]}(y_{j-1})$ by $f^{[i]}([y_{j-1}])$. Therefore,

$$[y_j] = [y_{j-1}] + \sum_{i=1}^{k-1} h_{j-1}^i f^{[i]}([y_{j-1}]) + h_{j-1}^k f^{[k]}([\tilde{y}_{j-1}]) \tag{19}$$

contains $y(t_j; t_0, [y_0])$. However, as explained below, (19) illustrates how replacing real numbers in an algorithm by intervals often leads to large overestimations.

Taking widths on both sides of (19) and using (11), we obtain

$$\begin{aligned} \mathrm{w}([y_j]) &= \mathrm{w}([y_{j-1}]) + \sum_{i=1}^{k-1} h_{j-1}^i \mathrm{w}\Big(f^{[i]}([y_{j-1}])\Big) + h_{j-1}^k \mathrm{w}\Big(f^{[k]}([\tilde{y}_{j-1}])\Big) \\ &\geq \mathrm{w}([y_{j-1}]), \end{aligned}$$

which implies that the width of $[y_j]$ almost always increases[2] with j, even if the true solution contracts.

A better approach is to apply the mean-value theorem to $f^{[i]}$ in (18), obtaining for any $\hat{y}_{j-1} \in [y_{j-1}]$

$$y_j = \hat{y}_{j-1} + \sum_{i=1}^{k-1} h_{j-1}^i f^{[i]}(\hat{y}_{j-1}) + h_{j-1}^k f^{[k]}(y; t_{j-1}, t_j)$$
$$+ \left\{ I + \sum_{i=1}^{k-1} h_{j-1}^i J\Big(f^{[i]}; y_{j-1}, \hat{y}_{j-1}\Big) \right\} (y_{j-1} - \hat{y}_{j-1}), \qquad (20)$$

where $J\Big(f^{[i]}; y_{j-1}, \hat{y}_{j-1}\Big)$ is the Jacobian of $f^{[i]}$ with its lth row evaluated at $y_{j-1} + \theta_{i,l}(\hat{y}_{j-1} - y_{j-1})$ for some $\theta_{i,l} \in [0,1]$, $l = 1, 2, \ldots, n$. This formula is the basis for the ITS methods of Moore [19], Eigenraam [8], Lohner [17], and Rihm [27]; see also [23].

Let

$$J\Big(f^{[i]}; [y_{j-1}]\Big) = \left\{ \left. \frac{\partial f^{[i]}(y)}{\partial y} \right|_{y=y_j} : \; y_j \in [y_j] \right\},$$

$$[S_{j-1}] = I + \sum_{i=1}^{k-1} h_{j-1}^i \, J\Big(f^{[i]}; [y_{j-1}]\Big), \quad \text{and} \quad [z_j] = h_{j-1}^k f^{[k]}([\tilde{y}_{j-1}]). \qquad (21)$$

The ith Jacobian $J\Big(f^{[i]}; [y_{j-1}]\Big)$ can be computed by generating the Taylor coefficient $f^{[i]}([y_{j-1}])$ and then differentiating it [2, 3]. Alternatively, these Jacobians can be computed by generating the Taylor coefficients for the associated variational equation [17].

Using (21), we can rewrite (20) as

$$[y_j] = \hat{y}_{j-1} + \sum_{i=1}^{k-1} h_{j-1}^i f^{[i]}(\hat{y}_{j-1}) + [z_j] + [S_{j-1}]([y_{j-1}] - \hat{y}_{j-1}). \qquad (22)$$

We refer to a method implementing (22) as the *direct method.*

[2] An equality is possible only in the trivial cases $h_{j-1} = 0$ or $\mathrm{w}\Big(f^{[i]}([y_{j-1}])\Big) = 0$, for all $i = 1, 2, \ldots k-1$, and $\mathrm{w}(f^{[k]}([\tilde{y}_{j-1}])) = 0$.

If we compute enclosures with (22), the widths of the computed intervals may decrease. However, this approach frequently works poorly, because the interval vector $[S_{j-1}]([y_{j-1}] - \hat{y}_{j-1})$ may significantly overestimate the set

$$\left\{ S_{j-1}(y_{j-1} - \hat{y}_{j-1}) \mid S_{j-1} \in [S_{j-1}],\ y_{j-1} \in [y_{j-1}] \right\}.$$

Such overestimations accumulate as the integration proceeds. This is often called the *wrapping effect*.

In Figure 1, we illustrate the wrapping of the parallelepiped $\{\, Ax \mid x \in [x] \,\}$ by the box $A[x]$, where

$$A = \begin{pmatrix} 1 & -2 \\ 3 & -4 \end{pmatrix}, \quad [x] = \begin{pmatrix} [0,1] \\ [0,1] \end{pmatrix}, \quad \text{and} \quad A[x] = \begin{pmatrix} [-2,1] \\ [-4,3] \end{pmatrix}.$$

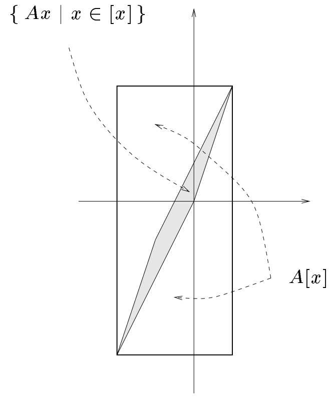

Fig. 1. The wrapping of the parallelepiped $\{\, Ax \mid x \in [x] \,\}$ by the box $A[x]$.

3.2.1 *The Wrapping Effect*

The wrapping effect is clearly illustrated by Moore's example [19],

$$y_1' = y_2, \quad y_2' = -y_1, \quad y(0) \in [y_0]. \tag{23}$$

The interval vector $[y_0]$ can be viewed as a box in the (y_1, y_2) plane. At $t_1 > 0$, the true solution of (23) is the rotated box shown in Figure 2. If we want to enclose this box in an interval vector, we have to wrap it by another box with sides parallel to the y_1 and y_2 axes. On the next step, this larger box is rotated and so must be enclosed in a still larger one. Thus, at each step, the enclosing boxes become larger and larger, but the true solution set is a rotated box of the same size as $[y_0]$. Moore showed that at $t = 2\pi$, the interval inclusion is inflated by a factor of $e^{2\pi} \approx 535$ as the stepsize approaches zero [19].

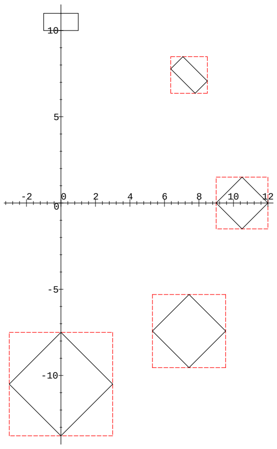

Fig. 2. The rotated box is wrapped at $t = \frac{\pi}{4}n$, $n = 1, 2, 3, 4$.

3.2.2 Lohner's Method

Here, we describe Lohner's QR-factorization method [16, 17], which is one of the most successful, general-purpose methods for reducing the wrapping effect.

Let

$$A_0 = I,\ \hat{y}_0 = \mathrm{m}([y_0]),\ \ [r_0] = [y_0] - \hat{y}_0, \quad \text{and} \tag{24}$$

$$\hat{y}_j = \hat{y}_{j-1} + \sum_{i=1}^{k-1} h_{j-1}^i f^{[i]}(\hat{y}_{j-1}) + \mathrm{m}([z_j]) \tag{25}$$

$(j \geq 1)$, where I is the identity matrix. From (20–21) and (24–25), we compute

$$[y_j] = \hat{y}_{j-1} + \sum_{i=1}^{k-1} h_{j-1}^i f^{[i]}(\hat{y}_{j-1}) + [z_j] + \Big([S_{j-1}]\, A_{j-1}\Big)[r_{j-1}] \tag{26}$$

and propagate for the next step the interval vector

$$[r_j] = \Big(A_j^{-1}([S_{j-1}]\, A_{j-1})\Big)[r_{j-1}] + A_j^{-1}\Big([z_j] - \mathrm{m}([z_j])\Big), \tag{27}$$

where $A_j \in \mathbb{R}^{n \times n}$ is nonsingular for $j = 1, 2, \ldots,$.

In Lohner's QR-factorization method,

$$
\begin{aligned}
\widehat{A}_j &= \mathrm{m}([S_{j-1}]A_{j-1}), \quad \widehat{A}_j = Q_j R_j, \quad \text{and} \\
A_j &= Q_j,
\end{aligned}
$$

where Q_j is orthogonal and R_j is upper triangular. Other choices for A_j are discussed in [16, 17, 20, 22].

One explanation why this method is successful at reducing the wrapping effect is that we enclose the solution on each step in a moving orthogonal coordinate system that "matches" the solution set; for more details, see [16, 17, 20, 23]. In section 6, we show that this choice for A_j ensures better stability of the QR-factorization method, compared to the direct method.

4 An Interval-Hermite Obreschkoff Method

For more than 30 years, Taylor series has been the only effective approach for computing rigorous bounds on the solution of an IVP for an ODE. Recently, we developed a new scheme, an interval Hermite-Obreschkoff (IHO) method [20, 21]. Here, we outline the method and its potential.

Let

$$
c_i^{q,p} = \frac{q!}{(p+q)!} \frac{(q+p-i)!}{(q-i)!} \tag{28}
$$

(q, p, and $i \geq 0$), $y_{j-1} = y(t_{j-1}; t_0, y_0)$, and $y_j = y(t_j; t_0, y_0)$. It can be shown that

$$
\begin{aligned}
\sum_{i=0}^{q} (-1)^i c_i^{q,p} h_{j-1}^i f^{[i]}(y_j) = &\sum_{i=0}^{p} c_i^{p,q} h_{j-1}^i f^{[i]}(y_{j-1}) \\
&+(-1)^q \frac{q!p!}{(p+q)!} h_{j-1}^k f^{[k]}(y; t_{j-1}, t_j),
\end{aligned} \tag{29}
$$

where $k = p + q + 1$ [20, 31].

The Hermite-Obreschkoff formula (29) is the basis of our IHO method. As in an ITS method, we can easily bound the local truncation error in the IHO method, since we can readily generate the interval Taylor coefficient $f^{[k]}([\tilde{y}_{j-1}])$.

The method we propose in [20] consists of two phases, which can be considered as a predictor and a corrector. The predictor computes an enclosure $[y_j^{(0)}]$ of the solution at t_j, and using this enclosure, the corrector computes a tighter enclosure $[y_j] \subseteq [y_j^{(0)}]$ at t_j. If $q > 0$, (29) is an implicit scheme. The corrector applies a Newton-like step to tighten $[y_j^{(0)}]$.

We have shown in [20, 21] that for the same order and stepsize, our IHO method has smaller local error, better stability, and requires fewer Jacobian evaluations than an ITS method. The extra cost of the Newton step is one matrix inversion and a few matrix multiplications.

5 Instability from the Formula for the Truncation Error

In this section, we investigate the stability of the ITS and IHO methods, when applied with a constant stepsize and order to the test problem

$$y' = \lambda y, \quad y(0) = y_0, \tag{30}$$

where λ and $y_0 \in \mathbb{R}$, and $\lambda < 0$. Since we have not defined complex interval arithmetic, we do not consider problems with λ complex. Note also that the wrapping effect does not occur in one-dimensional problems.

For the remainder of this paper, we consider ITS methods with a constant stepsize and order k truncation error and in this section, we consider the IHO scheme with a constant stepsize and order $k = p + q + 1$ truncation error.

5.1 The Interval Taylor Series Method

Suppose that at $t_{j-1} > 0$, we have computed a tight enclosure $[y_{j-1}^{\text{ITS}}]$ of the solution of (30) with an ITS method, and $[\tilde{y}_{j-1}^{\text{ITS}}]$ is an a priori enclosure of the solution on $[t_{j-1}, t_j]$ for all $y_{j-1} \in [y_{j-1}^{\text{ITS}}]$, where $[y_0^{\text{ITS}}] = y_0$.

Let

$$T_r(z) = \sum_{i=0}^{r} \frac{z^i}{i!}. \tag{31}$$

Using (31), an ITS method for computing tight enclosures of the solution to (30) can be written as

$$[y_j^{\text{ITS}}] = T_{k-1}(\lambda h)[y_{j-1}^{\text{ITS}}] + \frac{(\lambda h)^k}{k!}[\tilde{y}_{j-1}^{\text{ITS}}]; \tag{32}$$

cf. (20). Since $[\tilde{y}^{\text{ITS}}_{j-1}] \supseteq [y^{\text{ITS}}_{j-1}]$ and $\text{w}([\tilde{y}^{\text{ITS}}_{j-1}]) \geq \text{w}([y^{\text{ITS}}_{j-1}])$, we obtain from (31–32) that

$$\text{w}([y^{\text{ITS}}_{j}]) \geq \left(|T_{k-1}(\lambda h)| + \frac{|\lambda h|^k}{k!} \right) \text{w}([y^{\text{ITS}}_{j-1}]).$$

Therefore, if

$$|T_{k-1}(\lambda h)| + \frac{|\lambda h|^k}{k!} > 1, \tag{33}$$

the ITS method given by (32) is *asymptotically unstable*, in the sense that $\lim_{j\to\infty} \text{w}([y^{\text{ITS}}_{j}]) = \infty$. This result implies that we have restrictions on the stepsize not only from the function $T_{k-1}(\lambda h)$, as in point methods for IVPs for ODEs, but also from the factor $|\lambda h|^k/k!$ in the remainder term.

5.2 The Interval Hermite-Obreschkoff Method

Let $y_{j-1} \in [y^{\text{IHO}}_{j-1}]$, where we assume that $[y^{\text{IHO}}_{j-1}]$ is computed with an IHO method, and $[y^{\text{IHO}}_{0}] = y_0$. The formula (29) reduces to

$$\begin{aligned}\left(\sum_{i=0}^{q} (-1)^i c_i^{q,p} \frac{(\lambda h)^i}{i!} \right) y_j = {} & \left(\sum_{i=0}^{p} c_i^{p,q} \frac{(\lambda h)^i}{i!} \right) y_{j-1} \\ & + (-1)^q \frac{q!p!}{(p+q)!} \frac{(\lambda h)^{p+q+1}}{(p+q+1)!} y(\xi_{j-1}),\end{aligned} \tag{34}$$

where $\xi_{j-1} \in [t_{j-1}, t_j]$, and $c_i^{q,p}$ and $c_i^{p,q}$ are defined in (28). Define

$$R_{p,q}(z) = \frac{\sum_{i=0}^{p} c_i^{p,q} \frac{z^i}{i!}}{\sum_{i=0}^{q} c_i^{q,p} \frac{(-z)^i}{i!}} \quad \text{and} \quad Q_{p,q}(z) = \sum_{i=0}^{q} c_i^{q,p} \frac{(-z)^i}{i!}. \tag{35}$$

Also let $[\tilde{y}^{\text{IHO}}_{j-1}]$ be an a priori enclosure of the solution on $[t_{j-1}, t_j]$ for any $y_{j-1} \in [y^{\text{IHO}}_{j-1}]$. From (34–35), we compute an enclosure $[y^{\text{IHO}}_{j}]$ by

$$[y^{\text{IHO}}_{j}] = R_{p,q}(\lambda h)[y^{\text{IHO}}_{j-1}] + (-1)^q \frac{\gamma_{p,q}}{Q_{p,q}(\lambda h)} \frac{|\lambda h|^k}{k!} [\tilde{y}^{\text{IHO}}_{j-1}],$$

where $\gamma_{p,q} = q!p!/(p+q)!$. Then,

$$\mathrm{w}([y_j^{\mathrm{IHO}}]) \geq \left(|R_{p,q}(\lambda h)| + \frac{\gamma_{p,q}}{|Q_{p,q}(\lambda h)|} \frac{|\lambda h|^k}{k!} \right) \mathrm{w}([y_{j-1}^{\mathrm{IHO}}]).$$

Therefore, the IHO method is asymptotically unstable in the sense that $\lim_{j\to\infty} \mathrm{w}([y_j^{\mathrm{IHO}}]) = \infty$ for h satisfying

$$|R_{p,q}(\lambda h)| + \frac{\gamma_{p,q}}{|Q_{p,q}(\lambda h)|} \frac{|\lambda h|^k}{k!} > 1. \tag{36}$$

In (33) and (36),

$$T_{k-1}(z) = e^z + O(z^k) \quad \text{and} \quad R_{p,q}(z) = e^z + O(z^{p+q+1}) = e^z + O(z^k)$$

are approximations to e^z of the same order. In particular, $R_{p,q}(z)$ is the Padé rational approximation to e^z (see for example [26]). For the ITS method, $|T_{k-1}(\lambda h)| < 1$ if and only if λh is in the finite stability region of $T_{k-1}(z)$. However, for the IHO method with $0 > \lambda \in \mathbb{R}$, $|R_{p,q}(\lambda h)| < 1$ for any $h > 0$ if $q \geq p$ [30], and $|R_{p,q}(\lambda h)| \to 0$ as $\lambda h \to -\infty$ if $q > p$. Roughly speaking, the stepsize in the ITS method is restricted by both

$$|T_{k-1}(\lambda h)| \quad \text{and} \quad \frac{|\lambda h|^k}{k!},$$

while in the IHO method, the stepsize is limited mainly by

$$\frac{\gamma_{p,q}}{|Q_{p,q}(\lambda h)|} \frac{|\lambda h|^k}{k!}.$$

In the latter case, $\gamma_{p,q}/Q_{p,q}(\lambda h)$ is usually much smaller than one; thus, the stepsize limit for the IHO method is usually much larger than for the ITS method.

An important point to note here is that an interval version of a standard numerical method, such as the Hermite-Obreschkoff formula (29), that is suitable for stiff problems may still have a restriction on the stepsize. To obtain an interval method without a stepsize restriction, we must find a stable formula not only for the propagated error, but also for the associated truncation error.

6 A New Perspective on the Wrapping Effect

The problem of reducing the wrapping effect has usually been studied from a geometric perspective as finding an enclosing set that introduces as little overestimation of the enclosed set as possible. For example, parallelepipeds [8, 13, 17, 19], ellipsoids [11, 12, 25], convex polygons [29], and zonotopes [14] have been employed to reduce the wrapping effect.

In [22], we linked the wrapping effect to the stability of an ITS method for IVPs for ODEs and interpreted the problem of reducing the wrapping effect as one of finding a more stable scheme for advancing the solution. This allowed us to study the stability of several ITS methods (and thereby the wrapping effect) by employing eigenvalue techniques, which have proven so useful in studying the stability of point methods.

In this section, we explain first how the wrapping effect can cause instability in the direct method and then show that Lohner's QR-factorization method provides a more stable scheme for propagating the error.

6.1 The Wrapping Effect as a Source of Instability

Consider the IVP

$$y' = By, \quad y(0) = y_0, \tag{37}$$

where $B \in \mathbb{R}^{n\times n}$ and $n \geq 2$.

Let $T = T_{k-1}(hB)$. When applied with a constant stepsize h and order k to (37), the direct method (22) reduces to

$$[y_j] = T[y_{j-1}] + [z_j]. \tag{38}$$

Taking widths on both sides of (38) and using (11) and (13), we obtain

$$\mathrm{w}([y_j]) = |T|\mathrm{w}([y_{j-1}]) + \mathrm{w}([z_j]).$$

We can interpret $\mathrm{w}([y_j])$ as the size of the bound on the global error and $\mathrm{w}([z_j])$ as the size of the bound on the local error. We can keep $\mathrm{w}([z_j])$ small by decreasing the stepsize or increasing the order. However, $T[y_{j-1}]$ can be a large overestimation of the set $\{\, Ty_{j-1} \mid y_{j-1} \in [y_{j-1}] \,\}$. The reason for this is that $\{\, Ty_{j-1} \mid y_{j-1} \in [y_{j-1}] \,\}$ is not generally a box, but it is enclosed, or wrapped, by the box $T[y_{j-1}]$; cf. Figure 1. Note that we wish to compute $\{\, Ty_{j-1} \mid y_{j-1} \in [y_{j-1}] \,\}$, but this is generally infeasible. So we compute $T[y_{j-1}]$ instead. That is, we normally compute on each step products of the form $T[y_{j-1}]$, and, consequently, we may incur a wrapping on each step, resulting in unacceptably wide bounds for the solution of (37).

Consider now the point Taylor series (PTS) method given by

$$u_j = Tu_{j-1}, \quad u_0 = y_0. \tag{39}$$

If we denote by $\delta_j = y(t_j) - u_j$ the global error of this method at t_j, then

$$\delta_j = T\delta_{j-1} + \check{z}_j,$$

where $\check{z}_j$ is the local truncation error of this method.

An important observation is that the global error in the PTS method propagates with T, while the global error in the direct method propagates with $|T|$. Denote by $\rho(A)$ the spectral radius of $A \in \mathbb{R}^{n\times n}$. It is well-known that $\rho(T) \leq \rho(|T|)$. Assuming that T and $|T|$ can be diagonalized, we showed in [22] that

- if $\rho(T) = \rho(|T|)$, then the bounds on the global error in these two methods are almost the same, and the wrapping effect is not a serious difficulty for the direct method; and
- if $\rho(T) < \rho(|T|)$, then the global error in the direct method can be much larger than the global error in the PTS method, and the direct method may suffer significantly from wrapping effect.

Thus, we can associate the wrapping effect in an ITS method with its global error being significantly larger than the global error of the corresponding PTS method. In particular, if $\rho(T) < 1$ and $\rho(|T|) > 1$, the PTS method is stable, but the associated ITS method is likely asymptotically unstable in the sense that $\lim_{j\to\infty} \|\mathrm{w}([y_j])\| = \infty$. To reduce the wrapping effect, or improve the stability of an ITS method, we must find a more stable scheme for advancing the solution.

6.2 How Lohner's QR-factorization Method Improves Stability

When applied with a constant stepsize and order to (37), an ITS method incorporating Lohner's QR-factorization scheme, which together we refer to as *Lohner's method*, can be written as

$$[y_j] = T\hat{y}_{j-1} + \Big(TA_{j-1}\Big)[r_{j-1}] + [z_j] \quad \text{and} \tag{40}$$

$$[r_j] = \Big(A_j^{-1}TA_{j-1}\Big)[r_{j-1}] + A_j^{-1}\Big([z_j] - s_j\Big). \tag{41}$$

The transformation matrices $A_j \in \mathbb{R}^{n\times n}$ satisfy

$$TA_{j-1} = Q_j R_j \tag{42}$$

$$A_j = Q_j, \tag{43}$$

where $A_0 = Q_0 = I$, Q_j is orthogonal, and R_j is upper triangular.

Using (42–43), we write (40–41) in the following equivalent form:

$$[y_j] = T\hat{y}_{j-1} + \Big(Q_j R_j\Big)[r_{j-1}] + [z_j] \tag{44}$$

$$[r_j] = R_j[r_{j-1}] + Q_j^T\Big([z_j] - s_j\Big). \tag{45}$$

The interval vector $[r_j]$ in (45) can be interpreted as an enclosure of the global error (at t_j) that is propagated to the next step. Obviously, we are interested in keeping the overestimation in $[r_j]$ as small as possible.

Since Q_j is orthogonal, $|A_j^{-1}|\mathrm{w}([z_j]) = |Q_j^T|\mathrm{w}([z_j])$ is not much bigger than $\mathrm{w}([z_j])$, which we keep sufficiently small (by reducing the stepsize or increasing the order). Since

$$\mathrm{w}([r_j]) = |R_j|\mathrm{w}([r_{j-1}]) + |Q_j^T|\mathrm{w}([z_j]), \tag{46}$$

we can consider $|R_j|$ as the matrix for propagating the global error in the QR method. As we discuss below, the nature of the R_j matrices ensures that this method is normally more stable than the direct method.

A key observation here is that (42–43) is the *simultaneous iteration* (see for example [32]) for computing the eigenvalues of T. This iteration is closely related to Francis' QR algorithm [9] for finding the eigenvalues of T. To see this, let

$$S_j = Q_j^T T Q_j \quad \text{and} \quad \widehat{Q}_j = Q_{j-1}^T Q_j \quad (j \geq 1). \tag{47}$$

Note that $S_0 = T$ since $Q_0 = I$. Then,

$$S_{j-1} = \widehat{Q}_j R_j \quad \text{and} \tag{48}$$

$$S_j = R_j \widehat{Q}_j. \tag{49}$$

Now observe that the iteration (48–49) is just the *unshifted QR algorithm* for finding the eigenvalues of T.

6.2.1 Eigenvalues of Different Magnitudes

Let T be nonsingular with eigenvalues $\{\, \gamma_i \,:\, i = 1, 2, \ldots, n \,\}$ of distinct magnitudes. Then, from Theorem 3 in [9], under the QR iteration (48–49), the elements below the principal diagonal of S_j tend to zero, the moduli of those above the diagonal tend to fixed values, and the elements on the principal diagonal tend to the eigenvalues of T.

Using this theorem, we showed in [22] that $\lim_{j\to\infty} |R_j| = R$ for some upper-triangular matrix R with eigenvalues $\{\, |\gamma_i| \,:\, i = 1, 2, \ldots, n \,\}$. Thus,

$$\lim_{j\to\infty} \rho(|R_j|) = \rho(R) = \rho(T). \tag{50}$$

Using (50), we derived in [22] an upper bound for the global error of Lohner's method and showed that this bound is not much bigger than the bound for the global error of the corresponding PTS method (39), for the same stepsize and order. We also showed on several examples, that, at each t_j, the global error of Lohner's method is not much bigger than the global error of the PTS method.

6.2.2 Eigenvalues of Equal Magnitude

If T is nonsingular with p eigenvalues of equal magnitude, then the matrices S_j, defined in (47), tend to a block upper-triangular form with a $p \times p$ block on its main diagonal [10]. The eigenvalues of this block tend to these p eigenvalues.

If T is nonsingular with at least one complex conjugate pair of eigenvalues and at most two eigenvalues of the same magnitude, then S_j tends to a block upper-triangular form with 1×1 and 2×2 blocks on its diagonal. The eigenvalues of these blocks tend to the eigenvalues of T. Note that S_j does not converge to a fixed matrix, but the eigenvalues of its diagonal blocks converge to the eigenvalues of T.

For this case, we showed in [22] that, as $j \to \infty$,

- if T has a dominant complex conjugate pair of eigenvalues, then

  $$\rho(|R_j|) \geq \rho(T),$$

  and $\rho(R_j)$ oscillates; and

- if T has a unique real eigenvalue of maximal modulus, then we should generally expect that

$$\rho(|R_j|) \to \rho(T).$$

Since the matrices $|R_j|$ do not approach a fixed matrix, as in the case of eigenvalues of distinct magnitudes, deriving a bound for the global error in this case is more difficult. However, we illustrated with four examples in [22] that the global error in Lohner's method is not normally much bigger than the corresponding global error of the PTS method.

References

[1] G. Alefeld and J. Herzberger. *Introduction to Interval Computations.* Academic Press, New York, 1983.

[2] C. Bendsten and O. Stauning. FADBAD, a flexible C++ package for automatic differentiation using the forward and backward methods. Technical Report 1996-x5-94, Department of Mathematical Modelling, Technical University of Denmark, DK-2800, Lyngby, Denmark, August 1996.

[3] C. Bendsten and O. Stauning. TADIFF, a flexible C++ package for automatic differentiation using Taylor series. Technical Report 1997-x5-94, Department of Mathematical Modelling, Technical University of Denmark, DK-2800, Lyngby, Denmark, April 1997.

[4] M. Berz and K. Makino. Verified integration of ODEs and flows using differential algebraic methods on high-order Taylor models. *Reliable Computing*, 4:361–369, 1998.

[5] G. Birkhoff and R. S. Varga. Discretization errors for well-set Cauchy problems: I. *J. Math. and Phys.*, 44:1–23, 1965.

[6] G. F. Corliss and R. Rihm. Validating an a priori enclosure using high-order Taylor series. In G. Alefeld and A. Frommer, editors, *Scientific Computing, Computer Arithmetic, and Validated Numerics*, pages 228–238. Akademie Verlag, Berlin, 1996.

[7] B. L. Ehle. On Padé approximations to the exponential function and A-stable methods for the numerical solution of initial value problems. *SIAM J. Math. Anal.*, 4:671–680, 1973.

[8] P. Eijgenraam. *The Solution of Initial Value Problems Using Interval Arithmetic.* Mathematical Centre Tracts No. 144. Stichting Mathematisch Centrum, Amsterdam, 1981.

[9] J. G. F. Francis. The QR transformation: A unitary analogue to the LR transformation—part 1. *The Computer Journal*, 4:265–271, 1961/1962.

[10] J. H.Wilkinson. *The Algebraic Eigenvalue Problem.* Oxford Science Publications, Oxford, England, 1965.

[11] L. W. Jackson. Automatic error analysis for the solution of ordinary

differential equations. Technical Report 28, Dept. of Computer Science, University of Toronto, 1971.

[12] W. Kahan. A computable error bound for systems of ordinary differential equations. *Abstract in SIAM Review*, 8:568–569, 1966.

[13] F. Krückeberg. Ordinary differential equations. In E. Hansen, editor, *Topics in Interval Analysis*, pages 91–97. Clarendon Press, Oxford, 1969.

[14] W. Kühn. Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61(1):47–67, 1998.

[15] J. D. Lambert. *Computational Methods in Ordinary Differential Equations*. John Wiley & Sons, 1977.

[16] R. J. Lohner. Enclosing the solutions of ordinary initial and boundary value problems. In E. W. Kaucher, U. W. Kulisch, and C. Ullrich, editors, *Computer Arithmetic: Scientific Computation and Programming Languages*, pages 255–286. Wiley-Teubner Series in Computer Science, Stuttgart, 1987.

[17] R. J. Lohner. *Einschließung der Lösung gewöhnlicher Anfangs– und Randwertaufgaben und Anwendungen*. PhD thesis, Universität Karlsruhe, 1988.

[18] R. J. Lohner. Step size and order control in the verified solution of IVP with ODE's, 1995. SciCADE'95 International Conference on Scientific Computation and Differential Equations, Stanford University, Calif., March 28 – April 1, 1995.

[19] R. E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs, N.J., 1966.

[20] N. S. Nedialkov. *Computing Rigorous Bounds on the Solution of an Initial Value Problem for an Ordinary Differential Equation*. PhD thesis, Department of Computer Science, University of Toronto, Toronto, Canada, M5S 3G4, February 1999.

[21] N. S. Nedialkov and K. R. Jackson. An interval Hermite-Obreschkoff method for computing rigorous bounds on the solution of an initial value problem for an ordinary differential equation. *Reliable Computing*, 5(3):289–310, 1999. Also in T. Csendes, editor, *Developments in Reliable Computing*, pp. 289–310, Kluwer, Dordrecht, Netherlands, 1999.

[22] N. S. Nedialkov and K. R. Jackson. A new perspective on the wrapping effect in interval methods for initial value problems for ordinary differential equations, 2000. In preparation, 47 pages.

[23] N. S. Nedialkov, K. R. Jackson, and G. F. Corliss. Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105(1):21–68, 1999.

[24] N. S. Nedialkov, K. R. Jackson, and J. D. Pryce. An effective high-order interval method for validating existence and uniqueness of the solution of an IVP for an ODE, 2000. Accepted for publication in Reliable Computing, 17 pages.

[25] A. Neumaier. Global, rigorous and realistic bounds for the solution of dissipative differential equations. Part I : Theory. *Computing*, 52(4):315–

336, 1994.

[26] A. Ralston. *A First Course in Numerical Analysis.* McGraw-Hill, New York, 2nd edition, 1978.

[27] R. Rihm. On a class of enclosure methods for initial value problems. *Computing*, 53:369–377, 1994.

[28] H. Spreuer and E. Adams. On the existence and the verified determination of homoclinic and heteroclinic orbits of the origin for the Lorenz system. *Computing Suppl.*, 9:233–246, 1993.

[29] N. Stewart. A heuristic to reduce the wrapping effect in the numerical solution of $x' = f(t,x)$. *BIT*, 11:328–337, 1971.

[30] R. S. Varga. On higher order stable implicit methods for solving parabolic differential equations. *J. Math. and Phys.*, 40:220–231, 1961.

[31] G. Wanner. On the integration of stiff differential equations. In *Proceedings of the Colloquium on Numerical Analysis*, volume 37 of *Internat. Ser. Numer. Math.*, pages 209–226, Basel, 1977. Birkhäuser.

[32] D. S. Watkins. Understanding the QR algorithm. *SIAM Review*, 24(4):427–440, October 1982.