SUPPRESSION OF THE WRAPPING EFFECT BY TAYLOR MODEL- BASED VALIDATED INTEGRATORS MSU REPORT MSUHEP 40910

KYOKO MAKINO AND MARTIN BERZ

Abstract. The validated integration of ranges of initial conditons through ODEs faces two major challenges, namely the precise representation of the flow over the short term, and the avoidance of unfavorable buildup of errors in the long term. We discuss methods for meeting those challenges within the framework of Taylor model methods, in which the dependence on initial conditions is expressed by a high-order multivariate polynomial and a remainder bound. Numerous examples of the performance of the methods and comparisons to other approaches are given.

1. Introduction

The validated integration of differential equations is one of the important applications of interval- and related validated methods; in fact, the desire to integrate the dynamics of objects in the solar system has served as one of the original motivations for their development. Compared to other uses of validated methods, validated integration is particularly difficult because of the quite extended number of arithmetic operations and the fact that often similar operations repeat a great number of times.

While the problem of repeated application of similar operations manifests itself very clearly in validated integration, it is also affecting conventional integration, although in this case the effects are often more difficult to assess rigorously. Within the framework of conventional integration, the problems are usually tackled by using sufficiently small step size and methods of sufficient order for the step size under consideration, and optimizing the parameters of the algorithms as for example the coefficients in Runge Kutta and other frequently used tools.

However, the long-term control of these errors is much more difficult and represents a fundamental problem, in particular for nonlinear motion. This fact manifests itself particularly clearly in the frequent use of integration schemes that preserve certain known symmetries of the system like geometric constraints or symplecticity, because it is observed that conventional integrators do not satisfy these constraints well enough. It is hoped, then, that imposing the constraints leads to higher computational accuracy, an approach that is indeed often successful, but also often difficult to quantitatively assess.

Within the context of validated integration, the two primary concerns are that on the one hand, it is necessary to not only transport points, but rather regions, since even starting with points leads to regions. Already in his 1966 book[34] and in an

earlier paper[33], Moore describes this fundamental problem of the verified solution of differential equations of dimension 2 and higher, the need for re-packaging of the flow of the ODE with as little loss as possible, to avoid what is usually called the wrapping effect. On the other hand, there is less room for empirical approaches for the long term since they often do not lend themselves to obtaining rigorous tighter enclosures. Thus validated integration is faced with the need to address the following issues:

- Representing the flow accurately, i.e. providing a tight enclosure for the action of the differential equation on an extended region for a time step Δt
- Preventing local errors from accumulating in an unfavorable way when integrating over longer times.

To address the re-packaging or wrapping problem, which as observed by Moore [34][33] leads to an error that scales linearly with the step size Δt and hence cannot be controlled by merely refining the step size, Moore proposes to express the differential equations and its solution in a moving coordinate system, which entails that in this system, the solution set will always be nearly "upright" and thus is expected to be encloseable with intervals at much reduced loss. The coordinate system originally chosen by Moore is an approximation of the linearized solution that is first order in time and has the form

$$M_n = I + \Delta t_n \cdot f'(x_{n-1}, t_{n-1})$$

where I is the identity and f is the right hand side of the differential equation. The local coordinate system after step n is obtained recursively as

$$A_n = M_n \cdot A_{n-1}$$

and the enclosure of the solution is given by

$$r_n = A_n[r_0]$$

which is the linear transformation under the matrix A_n of the original box enclosing the set of initial conditions. Moore observed that if the solution is expressed in terms of the matrix A_n, the overestimation due to the need for re-packaging grows with Δt^2, and thus a reduction of the step size can effectively reduce the wrapping problem.

The method was further extended by Eijgenraam [10], who instead of M_n chooses matrices of the form

$$S_n = I + \sum_{i=1}^{k} \frac{\Delta t_n^i}{i!} f^{(i)}(x_{n-1}, t)$$

where again the local coordinate system after step n is given by $A_n = S_n \cdot A_{n-1}$. For larger Δt, the matrices S_n represents a better approximation of the linear transformation describing the propagation by the step Δt of the linearized ODE. The f_i are defined recursively as $f_i = f'_{i-1} \cdot f(x)$ and are also known as the Lie derivatives of the ODE, and k is a suitably large fixed value. This approach is often also referred to as the parallelepiped (PE) method.

In his ground breaking work on validated integration, Lohner [21][22][20][23][24][25] added another variant based on an orthogonal coordinate system that is obtained by using the QR-decomposition of the matrix A_n. Specifically, the columns of A_n are sorted in descending order by Euclidean length, and the transformation matrix is chosen as the orthogonal part Q_n of the QR decomposition of the matrix A_n.

While seemingly providing a less accurate approximation of the linearization than the propagation of S_n, the method has the significant advantage that the matrix Q_n is always well-conditioned by virtue of being orthogonal; thus the inversions necessary in propagating to the next time step can always be executed reliably, and propagation of interval vectors through the matrix does not lead to significant overestimation. In situations where integration over sufficiently large times is required and in which case the A_n can easily become ill-conditioned, this QR method offers a significant advantage.

An enhancement of the conventional QR method for large initial domain boxes is the combination of a parallelepiped to describe the bulk of the flow and a remainder expressed by the QR method as proposed by Lohner [22]. In our future study, this PEQR method will often serve as a reference for comparison.

While it is very difficult to assess the relative merits of these approaches in the general setting, for the special case of linear time-independent ODEs it is possible to provide a quantitative analysis of the behavior of the approaches. This work was pioneered by Nedialkov and Jackson [36][38], and it was seen through an eigenvalue analysis similar to what is done in the study of stability of conventional ODE solvers that the asymptotic behavior of the QR method is essentially the same as that of conventional non-validated integration schemes. Many practical examples also support this assessment, and [38] contains a rather representative collection of them.

However, for non-autonomous systems, the situation is different even in the linear case; for example, Kühn [19] provides a rather elementary example consisting of a sequence of n matrices that when applied repeatedly lead to exponential growth in the QR method, while the product of the n matrices is actually unity. We will revisit this topic again below.

Other enclosures for the flows of the ODEs besides the parallelepipeds of the PE and QR methods have also been studied. It seems natural to consider structures that are invariant under linear transformations, which aside from numerical inaccuracies allows to at least represent the solution sets of linear ODEs. The natural choices are ellipses, which appear in the work of Jackson [15][16][17], Kahan[18] and Neumaier[40], and convex polygons [42] as well as the related zonotopes [19]. The latter are linear transformations from $R^{m\cdot n}$ into R^n, where the higher dimensions are populated successively by assigning a new dimension to any error term that reaches a certain minimum size; apparently the approximation becomes better and better the larger the parameter m is chosen.

From a formal point of view, the zonotope methods are interesting because not only are they invariant under linear transformation, but also under addition, which facilitates the use of the objects in arithmetic. The latter methods have the advantage that using proper strategies of how new faces are added and others removed from the object, error growth can be substantially slowed. Particularly fruitful approaches seem to be the attempts at finding the "smallest" polygon including an interval box in [42] and the cascade algorithm presented in [19].

Other methods of avoiding potential exponential error growth for linear systems are developed by Gambill and Skeel [13] using odd-even reduction of the $(Mn)\times(Mn)$ matrix propagating the M initial and intermediate conditions in the n-dimensional system, as well as the intuitive approach by Barbarosie[2] based on

propagating boundaries of sets instead of sets themselves, which can be beneficially applied to two-dimensional problems.

All methods based on families of invariants of linear transformations discussed above, namely the PE, QR, PEQR, ellipsoid, and zonotope methods, have the following properties:

- the enclosure sets for the flow are convex, while nonlinear problems may require non-convex sets
- the accuracy of the enclosure, measured by the interval remainder bound, scales at most quadratically with size for nonlinear problems
- the families are not invariant under nonlinear transformations.

The Taylor model-based integrator introduced in [26] overcomes these three difficulties; relationships between the coordinates $x(t)$ and initial coordinates x_i are expressed in terms of a Taylor model [26][27][31][32] (P, I) consisting of a polynomial with floating point coefficients $P : R^n \to R^n$ and an n-dimensional interval I, both of which depend on t, such that $x(t) \in P(x_i) + I$. The representation of final coordinates in terms of initial coordinates in terms of Taylor models has the properties

- the enclosure sets can be either convex or concave
- the accuracy of the enclosure scales with order $n + 1$, where n is the order of the Taylor models being used
- the family of Taylor models is invariant under nonlinear transformations

More specifically, the Taylor model method [27, 26] combines interval methods for validation and high order automatic differentiation for efficient modeling of local functional behavior. The method represents a multivariate functional dependence f in the domain B by a high order multivariate Taylor polynomial P and the remainder bound interval I as

$$f(x) \in P(x - x_R) + I \text{ for all } x \in B, \tag{1.1}$$

where x_R is the reference point of the Taylor expansion. The nth order Taylor polynomial P is expressed by floating point coefficients, and it captures the bulk of functional dependency. Because the manipulation of those polynomials can be performed by operations on the coefficients where the minor errors due to their floating point nature are moved into the remainder bound, the major source of interval overestimation is removed, and overestimation only occurs in the remainder bound, the size of which scales with order n of the width of the domain[32].

The standard binary operations and intrinsic functions on Taylor models were implemented in the code COSY Infinity [26, 4]. For the treatment of ODEs, it is of particular significance that the antiderivation operation ∂^{-1} can be treated as an intrinsic function in the Taylor model structure [26, 6]. This formally removes the difference between the solution of ODEs and merely algebraic equations based on fixed point methods.

When applied to the verified integrations of ODEs [5], the following advantages have been observed.

- The inclusion requirement asserting existence of a solution reduces to a mere inclusion of the remainder intervals, and different from conventional methods based on two separate algorithms for initial validation by an Euler step and subsequent higher order execution, the entire steps is performed

in one algorithm. There is also no need to utilize additional ODEs for derivatives.
- The direct availability of the antiderivation on Taylor models allows to treat the Picard operator like any other function, avoiding the need to explicitly bound error terms of integration formulas and leading to a rather straightforward validated fixed point problem.
- The explicit dependency on initial variables is carried through the whole integration process. This controls the bulk of the dependency problem very efficiently and hence the main source of wrapping effect is eliminated to order $n+1$.

The results of the methods developed in [5] can be summarized in the following theorem.

Theorem 1. ***(Continuous Dynamical System with Taylor Models)*** *Let $P+I$ be an n-dimensional Taylor model describing the flow of the ODE at the time t; i.e. for all initial conditions x_0 in the original domain region $B \subset R^n$, we have*

$$x(x_0,t) \in I + \bigcup_{x_0 \in B} P(x_0).$$

Let $P^*(x_0,t)$ be the invariant polynomial depending on x_0 and t obtained in [5]*, and assume that the self-inclusion step of the Picard Operator mapping described there is satisfied over the interval $[t, t+\Delta t]$ by the remainder bound I^*. Then for all $x_0 \in B$, we have*

$$x(x_0,t+\Delta t) \in I^* + \bigcup_{x_0 \in B} P^*(x_0,t+\Delta t).$$

Furthermore, if even $x\,(x_0,t) \in P\,(x_0)+I$, then $x\,(x_0,t+\Delta t) \in P^*(x_0,t+\Delta t)+I^*$.

By induction over the individual steps, we obtain a relationship between initial conditions and final conditions at time t. Thus formally, the continuous case is made equivalent to the discrete case, for which the respective property follows immediately from the respective enclosure properties of Taylor models, as described for example in [32].

Theorem 2. ***(Discrete Dynamical System with Taylor Models)*** *Let $P+I$ be an n-dimensional Taylor model describing the flow of the discrete dynamical system $x_{n+1} = f(x_n,n)$, i.e. for all initial conditions x_0 in the original domain region $B \subset R^n$, we have*

$$x_n(x_0) \in I + \bigcup_{x_0 \in B} P(x_0).$$

Let $P^* + I^*$ be the Taylor model evaluation of $f\,(P+I,n)$. Then for all $x_0 \in B$, we have

$$x_{n+1}(x_0) \in I^* + \bigcup_{x_0 \in B} P^*(x_0).$$

Furthermore, if even $x_n\,(x_0,t) \in P\,(x_0)+I$, then $x_{n+1}\,(x_0) \in P^*(x_0)+I^*$.

The two theorems thus allows the validated study of continuous and discrete dynamical systems, provided that the Taylor model arithmetic is performed in a validated manner. In the case of the implementation in COSY, all errors in the floating point coefficients are fully accounted for [32][41].

For the purpose of practical efficiency, it is important that the treatment of the coefficients arithmetic supports sparsity, i.e. only coefficients that are nonzero (or more specifically, above a pre-specified accuracy threshold [32][41]) contribute to computational effort. Finally, for high dimensional systems and high expansion order n in time, one often observes that the expansion in the initial conditions does not need to be executed to the same order unless the dimensions of the original domain box is of a comparable size as the time step. This can be exploited simply by not setting the initial Taylor model to a linear form $P(x_0) = A \cdot x_0$ describing the original box, but rather choose $P(x_0) = A \cdot x_0^w$ for some suitable odd integer power w. In this way, throughout the computation, only powers of x_0 that are multiples of w appear, which effectively limits the expansion in initial conditions to the largest m that satisfies $m \cdot w \leq n$. Combined with sparsity methods, this can drastically reduce computational expense and storage requirements.

Definition 1. ***(Transversal Weighting)*** *Let the continuous dynamical system under consideration have v variables, and let the time expansion be executed to order n. Assume the initial box of interest is described by the Taylor model $P(x_0) = Ax_0^w$ where $w < n$ is an odd integer; then w is called the weighting of the transversal expansion.*

In a typical nonlinear problem one often finds that, already expansion order 3 or 5 in initial conditions allows the treatment of rather large initial domain boxes, while an expansion order of $n = 17$ in time may be desireable; an example of this can be seen below in figure4. This can be achieved by setting $w = 5$ or $w = 3$, respectively. Furthermore, in the case of linear ODEs where the dependency of final conditions on initial conditions is always linear, one can choose w in such a way that $2 \cdot w > n$, and thus only first order is retained. For the example case of $n = 17$, one may for example choose $w = 9$.

The method also has the interesting side effect that the effective expansion order in time of the higher order terms in the initial conditions is reduced, which because of their reduced importance and leads to additional computational savings without loss of accuracy. For example, in the $n = 17$ and $w = 5$ case, the first order dependence in initial condition is expanded to order 12, while the third order dependencies, of which there are many, are expanded only to order 2. From the combinatorial arguments in [3] it follows that the number of possible coefficients of order n in v initial conditions with weighting factor w is given by

$$N(n, v, w) = \sum_{j=0}^{[n/j]} \frac{(j+v-1)!}{j! \cdot (v-1)!} \cdot (n - w \cdot j + 1)$$

where $[x]$ denotes the Gauss bracket of x, the smallest integer not exceeding x. On the other hand, the number of floating point numbers necessary in a code like AWA that solves the ODE for the flow of the reference point and the first partials using polynomials with interval coefficients is $(n+1) \cdot (v+1) \cdot 2$.

For the purpose of providing some examples, we list in table 1 the number of floating point coefficients in a Taylor model of order n in v variables and with weighting w under the assumption of lack of any sparsity, i.e. all coefficients appear and lie above the accuracy threshold. The quantity n_i is the order of expansion in initial conditions. For comparison, the number of coefficients necessary to store

Order n	Variables v	Weighting w	Order n_i	Cosy Coefs	AWA Coefs
17	3	9	1	41	144
17	5	9	1	57	216
17	10	9	1	97	396
17	20	9	1	177	756
17	3	5	3	135	144
17	5	5	3	308	216
17	10	5	3	1248	396
17	20	5	3	6578	756
13	5	3	4	504	168
13	10	3	4	3094	308
15	5	3	5	882	192
15	10	3	5	7098	352

TABLE 1. Number of floating point numbers necessary to store all appearing partial derivaties in COSY to order n_i in initial conditions, and in the first order code AWA

all interval endpoints of the $n_i = 1$ representation used in AWA is also given. The first four rows show the situation for the case most similar to the perfomance of the $n_i = 1$ case of AWA; the smaller number of COSY coefficients is due to the fact that on the one hand, instead of interval coefficients only real numbers are stored, and on the other hand that the expansion order in time for the dependence on initial condtions is reduced. The other rows show the situation for other choices of weights, which of course is more expensive; yet in the COSY scheme third order n_i at least for low dimensions can still be achieved with a similar number of coefficients of AWA.

In the following section, we will study in detail the two fundamental questions of validated integration, the accurate representation of flows of ODEs, and methods to prevent growth of the remainder bound, and illustrate the behavior with a large number of examples.

2. Faithful Representation of Flows by Taylor Models

As discussed in the previous section, the successful use of validated methods requires on the one hand the accurate representation of the solution sets over short time scales, and on the other hand the ability to suppress the long-term build up of errors. In this section we study the behavior of the Taylor model method with respect to the first question, which is directly connected to and characteristic of the mathematical behavior of the ODE being studied. We observe that for linear systems, this first source of errors is particularly easy to control, since the flows of linear ODEs are merely linear transformations of the initial coordinates. However, as simple as the matter is for linear ODEs, as complicated it is for nonlinear ODEs. In this case, except for special cases there is no simple representation of the dependency of final conditions on initial conditions. This is the prime reason why nonlinear ODEs represent the real challenge in the validated integration of differential equations, and results obtained for the purely linear case are often not characteristic for the behavior in nonlinear cases.

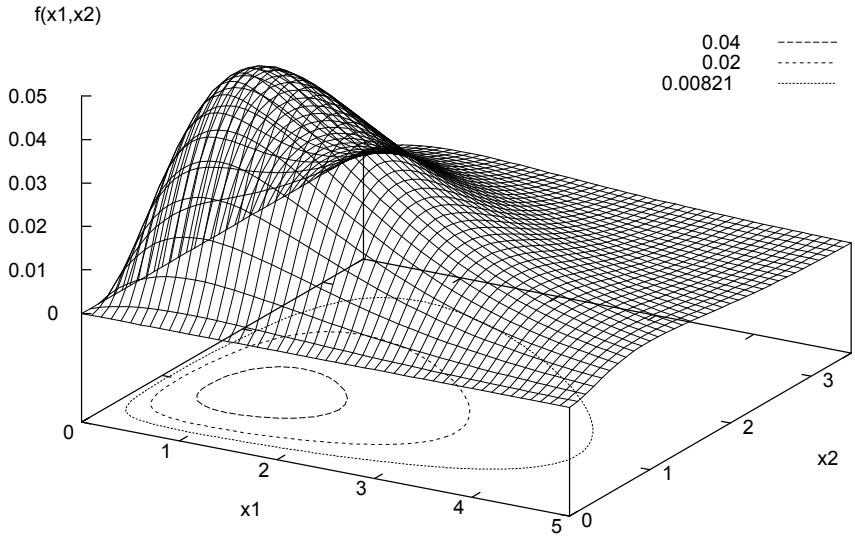

FIGURE 1. The solution trajectories of the Volterra equations, $f(x_1, x_2) \equiv x_1 x_2^2 e^{-x_1-2x_2} = c$(onstant). $f(x_1, x_2)$ is shown by mesh, and the contour lines for various values of c's are shown. The initial condition $(x_{01}, x_{02}) = (1, 3)$ corresponds to $c = 9e^{-7} \simeq 0.00821$.

2.1. **Examples - The Volterra Equations.** In the following, we illustrate the behavior of the Taylor model based integration scheme[5] and compare it to other methods, specifically the code AWA [20] as a representative of the conventional methods. We compare with COSY-VI, the (V)alidated (I)ntegrator based on the COSY language system [4] that is using the Taylor model arithmetic discussed in [32] [4].

The ODEs under consideration are the Volterra equations governing the growth of two conflicting populations, modeling a predator-prey relation, which are frequently used in the study of ODE solvers [1] [35]. The solution trajectories obey the constraint

$$C(x_1, x_2) = x_1 x_2^2 e^{-x_1-2x_2} = \text{Constant},$$

as can be seen by simple differentiation and insertion of the ODE, and thus the solutions follow the contour lines of the function C. In the quadrant characterized by $x_{1,2} > 0$, the constant is positive, which entails that contour lines of C cannot cross the x_1 or x_2 axis, and so contour lines originating in this quadrant stay in it. Furthermore, within this quadrant the function asymptotically approaches zero as x_1 or x_2 become large, and so contour lines are bounded and follow closed curves. Figure 1 illustrates the shape of C and a few of its contour lines. The period of one cycle of the solution depends on the initial condition, and outer orbits take longer.

The Volterra equations are a frequently cited example for the numerical verification of ODE solvers. For validated ODE solvers, their nonlinearity combined with their periodicity allows for a particularly transparent study of the wrapping effect.

We take the same model discussed by Ames and Adams [1] and by Moore [35], and have the initial condition interval vector centered around the point values used in their discussions. We aim, in such a way, to provide a good comparison between our approach and other approaches. The ODEs and initial conditions for the Volterra equations are

$$\begin{aligned} &\frac{dx_1}{dt} = 2x_1(1-x_2), \quad \frac{dx_2}{dt} = -x_2(1-x_1) \\ &x_{01} \in 1 + [-0.05, 0.05], \quad x_{02} \in 3 + [-0.05, 0.05] \quad \text{at } t = 0. \end{aligned} \tag{2.1}$$

The right hand side of the ODEs has the form of a "single use expression" (SUE), so it has no source of overestimation of arithmetic nature; this makes any overestimation due to the wrapping effect more clearly visible and separates this effect from the ability of the Taylor models to significantly reduce any dependency problem that may be present in the right hand side[28].

The solution trajectory for the point initial values $(x_{01}, x_{02}) = (1, 3)$ is a closed orbit with a period of about $T \simeq 5.488138468035$. We attempted to carry out the integration of the system with AWA and COSY-VI for one period T. As will be shown, the system starts to exhibit noticeable nonlinearity around $t \sim 4$.We used AWA in its standard mode; namely we use the enclosure method 4 based on an intersection of interval-vector and QR-decomposition [20, 22, 36]. AWA's error tolerances E_a and E_r, the absolute and the relative accuracy of the solution used for the step size control, are set at 10^{-12} each. However, those accuracy requirements are not necessarily achieved [20], as we will see later. The computational order has to be pre-set in both AWA and COSY-VI, and the same order was used to facilitate comparison. Both AWA and COSY-VI have automatic step size control, and it was observed that their choices of step sizes for different times t were similar. We performed the integration of the Volterra equations by AWA and COSY-VI with various computational orders, demanding the completion of one period T.

The pictures in Figure 2 show the solution regions $R(t)$ at various characteristic times, as they are enclosed by Taylor models. They are made based on the observation that flows of ODEs are bijective and thus the outer edges of the original box are mapped into the outer edges of the result after application of the ODE. Hence it is only necessary to draw four curves, two for which x_1 is fixed at the positive and negative values and x_2 varies, and two for which x_2 is fixed at the positive and negative values and x_1 moves. The remainder bounds are so small that they are insignificant to printer resolution.

Initially nonlinearity is not very significant, and until the nonlinearity becomes noticeable around $t \sim 4$, the solution regions $R(t)$ are still well represented by parallelepipeds. After that, the nonlinearity becomes larger and larger, and the solution region $R(t = 4.85)$ shows clear limitations to any attempt to accurately model the region by a parallelepiped or any other convex object. The nonlinearity temporarily decreases afterwards, but the strong nonlinearity returns just before the completion of the period as observed in $R(t = 5.45)$.

The solution enclosures at each time step of the 18th order Taylor model computation by COSY-VI are placed along the center point trajectory in Figure 3. Since COSY-VI completes the whole integration period without noticeable overestimation, it tightly keeps the closed orbit structure of the ODE trajectory. An elongation of the solution region $R(t)$ along the trajectory is observed, which is the result of different cycle periods for the various closed orbits. The dashed boxes are

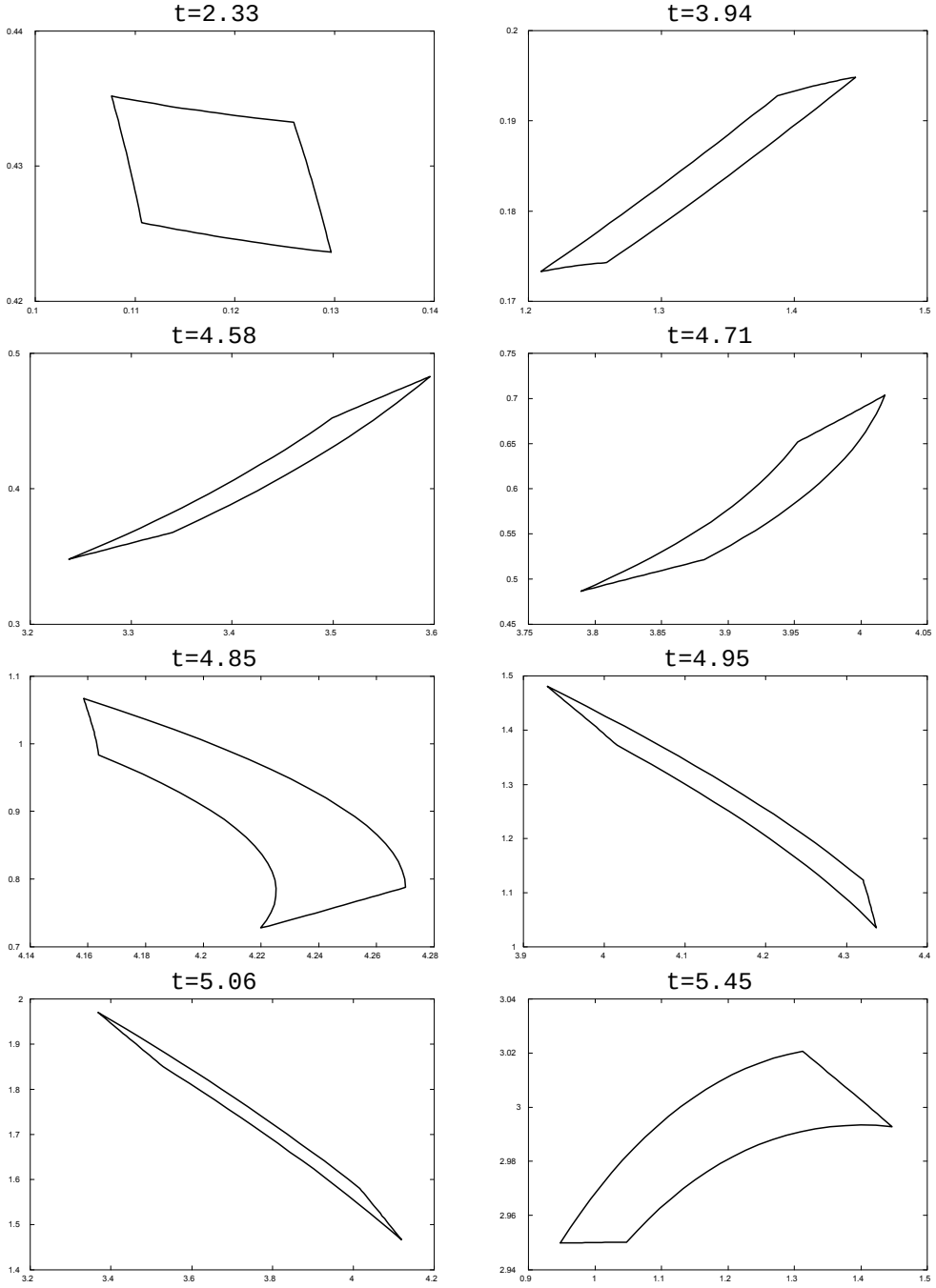

FIGURE 2. Solution enclosures at characteristic times, obtained by COSY-VI with computation order 18.

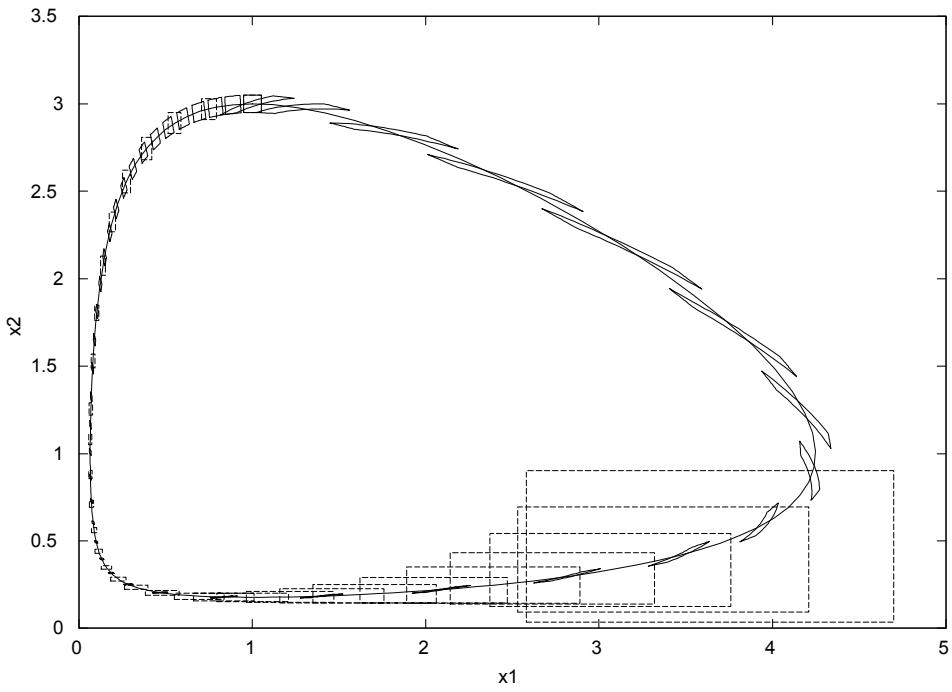

FIGURE 3. Solution enclosures of the Volterra eqs. at each time step by Taylor models (solid regions) and AWA (dashed boxes) in an 18th order computation.

the solution enclosure interval vectors obtained by AWA, showing the beginning of breakdown before $t = 4$. The last solution interval box by AWA in Figure 3 is at time $t \simeq 4.634$. In the case of AWA, despite of the error tolerance demand, a quick error growth is clearly observed after $t = 4$, and eventually integration cannot proceed despite drastic attempts at decreasing the step size. Eventually the box size reaches more than 10^{14} at time $t \simeq 4.93115$ and execution terminates. The dramatic growth in solution interval box size shows a clear correlation to the strong nonlinearity, which becomes apparent at $t = 4.85$ in Figure 2.

On the other hand, COSY-VI continues the computation during the period of strong nonlinearity by keeping the step size smaller; once the nonlinearity becomes weak again, the step size increases again. When the step size control is done only connected to the local error, the step size progress directly reflects the difficulty of integration due to the strength of nonlinearity.

The performance was studied with different computation orders for the system, but AWA terminated prematurely at nearly the same time regardless of the integration order; a typical consequence of the wrapping effect, which cannot be controlled by increasing the order. COSY-VI completed the whole demanded integration period T without difficulty when the expansion order in time was sufficiently high. For lower time expansion order, it was necessary to keep the step size small as mentioned earlier.

Order	COSY-VI	AWA	
	CPU time	CPU time	Breakdown time t
12	3.2 sec	13.6 sec	5.06039
18	13.6sec	10.7 sec	4.93115

TABLE 2. CPU time. COSY-VI completed the whole integration period $T = 5.488138468035$, but AWA broke down at time t.

Also listed in Table 2 is the CPU time comparison, using a 450 MHz Pentium III PC running Linux; the weighting w was chosen to be 1. Since AWA did not complete the period, we also listed the breakdown time t in the ODE system.

To illustrate the performance of the computation with COSY-VI, we now list the resulting Taylor model for the variables x_1 after the completion of one full cycle at $t = 5.488138468035000$. Shown are the floating point coefficients for each monomial, as well as its order and the exponents of the expansion in the initial conditions. Note that there is a third column for exponents, which during the integration step is used to describe the dependence on time, but which does not appear at the end since the final value of t is plugged in. We show all terms up to order 4, as well as the end of the expansions which contain terms of order 12, as well as the remainder bounds.

```
 I  COEFFICIENT             ORDER EXPONENTS
 1   1.000000000415308        0   0 0  0
 2  0.5000000002077984E-01    1   1 0  0
 3  0.1593548597307891        1   0 1  0
 4  0.2987903619745317E-02    2   2 0  0
 5  0.7967742985213962E-02    2   1 1  0
 6  0.1745863785938967E-01    2   0 2  0
 7  0.4979839364267220E-04    3   3 0  0
 8  0.5551021323566726E-03    3   2 1  0
 9  0.6348634118140111E-03    3   1 2  0
10  0.1191291279313411E-02    3   0 3  0
11  0.3258832737600261E-05    4   4 0  0
12  0.3241341493295573E-06    4   3 1  0
13  0.3862783708476137E-04    4   2 2  0
14  0.2689662801524732E-05    4   1 3  0
15  0.3564904350045831E-04    4   0 4  0
...
79  0.2264828694386490E-15   12  12 0  0
80  -.1070762043111673E-14   12  11 1  0
81  0.3189161647800073E-14   12  10 2  0
82  0.1429170282664684E-14   12   9 3  0
83  0.1168048490492948E-13   12   8 4  0
84  0.6197159510359881E-13   12   7 5  0
85  0.6886774467995614E-13   12   6 6  0
86  0.2141863127503214E-12   12   5 7  0
87  0.1915198148620145E-12   12   4 8  0
```

```
88  0.2264491972426495E-12  12   3 9  0
89  0.1788727621438823E-12  12   210  0
90  0.5499818896261770E-13  12   111  0
91  0.6996138986393415E-13  12   012  0
R     [-.1481801093188394E-008,0.1490922875566877E-008]
```

To understand the meaning of the terms, consider some examples. After one revolution, the center point of the first variable is mapped back to a value near 1, as expected. The coefficient describing the linear dependence of the final first variable on the initial first variable is around 0.05, corresponding to the original box width. There is also a linear dependence on the second variable of about 0.16, describing a substantial shearing of the end result, which is also clearly visible in figure 3. Furthermore, there are many higher order contributions; for example the second order dependence on x_1x_2 is around -0.00597, indicating an appreciable curvature, which is also noticeable in figure 3. The terms of order 12 are smaller than 10^{-12}, illustrating that the expansion of final conditions on initial conditions does indeed converge. The remainder bound has a width of about $3 \cdot 10^{-9}$, which is more than seven orders of magnitude less than the dependence on linear terms.

For the purpose of a more quantitative study of the behavior of the integrators, let us now consider in detail the execution of a single step of the integration process. We choose a region in which nonlinearity is sufficiently strong so that the effects can be noticed in one step. We choose as initial condition the linear part of the Taylor model at $t = 4.85$. Since AWA can not treat in detail the nonlinear solution set produced by COSY for this time, we delete its nonlinear terms and obtain an approximation of the solution set at the time of interest that has the form of a parallelepiped.

Then we use this parallelepiped to perform a single time step by the time Δt. We execute the step with COSY so that as a result, nonlinear terms are being populated. To simulate the behavior of AWA, all the resulting nonlinearities as well as the $(n+1)$st order remainder interval produced by COSY are bounded into an interval, which is a measure of the one-step accuracy of a linear code like AWA. It is likely that this estimation is somewhat optimistic since it ignores any possible dependency in the iterative process of the solution of the ODE.

Figure 4 shows the width of the resulting higher order terms as a function of the expansion order for various different time steps, where T is the time step recommended by COSY's step size controller. As can be seen, at order 1 the one-step error is around 10^{-3}, while for the smaller step sizes, between orders 4 to 6 the one-step error can be suppressed below 10^{-13}. Because of the high order dependence of the integration error on step size, the error at twice the recommended step size reaches only around 10^{-8}.Thus for a suitable step size, the one-step integration error produced by COSY's Taylor model method is 10 orders of magnitude less than that for a linear method.

It is also illuminating to study the behavior of the error as a function of the size of the parallelepiped. For this purpose we execute a step at the recommended step size for parallelepipeds scaled by various factors and observe the behavior at different orders. Figure 5 shows the resulting widths of the remainder bounds. All

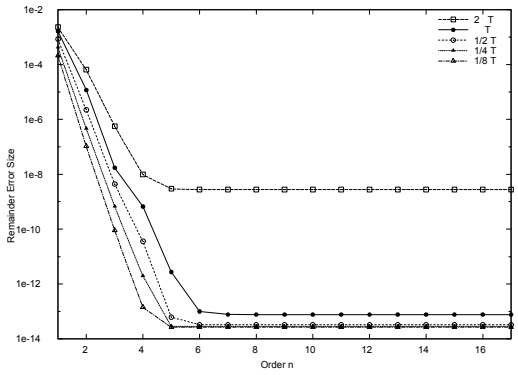

FIGURE 4. Remainder errors for a single step as a function of order and step size

boxes up to the original size of the box can be integrated to an accuracy below 10^{-13} for sufficiently high orders between 4 and 6; the larger box allows integration only to an error of 10^{-12}. On the other hand, a linear method similar to the one used in AWA can produce a one-step error only in the range of 10^{-2} to 10^{-5}. So altogether, again the Taylor model approach leads to a reduction of the one-step error by 7 to 10 orders of magnitude. Overall we observe that the Taylor model method has the

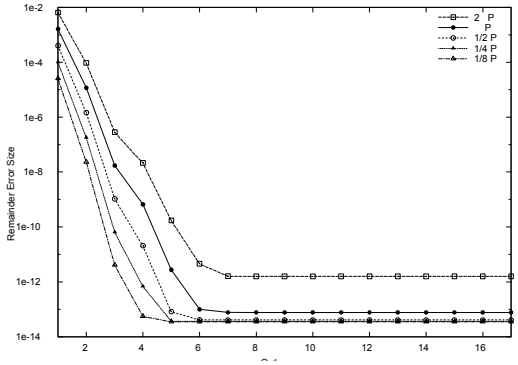

FIGURE 5. Remainder errors for a single step as a function of order and box size

ability to represent the solution set very accurately up to the error of the remainder bound, the size of which at a fixed time can be affected by the order of expansion in the transversal variable, as well as of course by the step size and as necessary the floating point accuracy. In fact, under the assumption of expandability of the flow in time and transversal variables, and under the assumption of arbitrary precision arithmetic, for a fixed t, the Taylor model method allows to represent the flow to any pre-specified accuracy.

3. Shrink Wrapping

In this section, we address one method to control the long-term growth of integration errors. As we saw in the last section, for a fixed time t of interest, the errors appearing in the remainder interval can at least in principle be kept as small as desired. However, for large values of the time t, the approach used there may become computationally impractical, and so it is desireable to develop schemes that limit the error growth as a function of time for a fixed expansion order and computational accuracy. The shrink wrapping method[30] is one approach for this purpose. It is based on the idea of enclosing the remainder error including floating point errors and errors due to the finite order in time within the range of the polynomial part of the Taylor model. By doing so, the remainder error ceases to be an interval, and instead is transformed into a variable that is retained explicitly up to the order of the Taylor model.

While in the linear case, this problem reduces to mere linear algebra, in the nonlinear case the situation is more involved, as the present nonlinear terms should not be also simply lumped into the linear parts at the same time; so the task requires to absorb the interval into a nonlinear structure, and we refer to it as shrink wrapping. In the following, we present one method to perform shrink wrapping; we point out that there are many variants of this approach, and while the one shown here is one of the simpler ones to outline, it is not necessarily the optimal choice for given problems.

As discussed in the introduction, after the kth step of the integration, the region occupied by the final variables is given by the set

$$A = I_0 + \bigcup_{x_0 \in B} \mathcal{M}_0(x_0), \tag{3.1}$$

where x_0 are the initial variables, B is the original box of initial conditions, $\mathcal{M}_0$ is the polynomial part of the Taylor model, and I_0 is the remainder bound interval; the sum is the conventional sum of sets. In the case of the COSY-VI integration, the map $\mathcal{M}_0$ can be scaled such that the original box B is unity, i.e. $B = [-1, 1]^v$. We assume this to be the case for the rest of the discussion. The remainder bound interval I_0 accounts for the local approximation error of the expansion in time carried out in the kth step as well as floating point errors and potentially other accumulated errors from previous steps; it is usually very small. As stated earlier, the purpose of shrink wrapping is to "absorb" the small remainder interval into a set very similar to the second part of the right hand side in eq. (3.1) via

$$A \subset A^* = I_0^* + \bigcup_{x_0 \in B} \mathcal{M}_0^*(x_0),$$

where $\mathcal{M}_0^*$ is a slightly modified polynomial, and I_0^* is a significantly reduced interval of the size of machine precision.

As the first step, we extract the constant part a_0 and linear part $M_0 \cdot x$ of $\mathcal{M}_0$ and determine a floating point approximation $\bar{M}_0^{-1}$ of the inverse of M_0. In case the ODEs admit unique solutions, as is typically the case for the problems at hand, also the linear part of the flow is invertible. Within a floating point environment, thus the attempt to invert the linear transformation M_0 will likely succeed as long as the linear transformation is sufficiently well-conditioned. If this is not the case, additional steps may be necessary, which will be discussed in some detail below.

After the approximate inverse $\bar{M_0^{-1}}$ has been determined, we apply the linear transformation $\bar{M_0^{-1}} \cdot (x - a_0)$ from the left to the Taylor model $\mathcal{M}_0(x_0) + I_0$ that describes the current flow. As a result, the constant part of the resulting Taylor model now vanishes, and its linear part is near identity. We write the resulting Taylor model as

$$\mathcal{M} + I = \mathcal{I} + \mathcal{S} + I,$$

where $\mathcal{I}$ is the identity, and the function $\mathcal{S}$ contains the nonlinear parts of the resulting Taylor model as well as some small linear corrections due to the error in inversion. We include I into the interval box $d \cdot [-1, 1]^v$, where d is a small number.

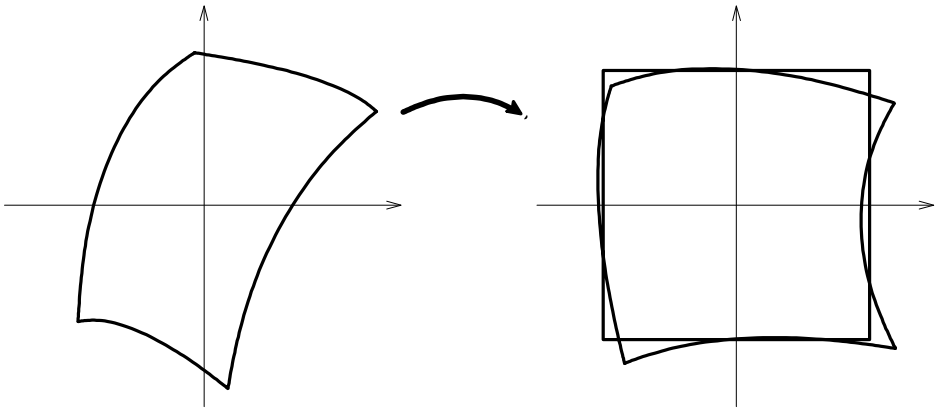

FIGURE 6. The region described by the Taylor model $\mathcal{M}_0 + I_0$ is transformed to be normalized as $\mathcal{I} + \mathcal{S} + I$, where $\mathcal{I}$ is the identity.

Definition 2. *Let $\mathcal{M} = \mathcal{I} + \mathcal{S} + I$, where $\mathcal{S}$ is a polynomial and I is a small interval. We include I into the interval box $d \cdot [-1, 1]^v$. We pick numbers s and t satisfying*

$$s \geq |\mathcal{S}_i(x)| \ \forall \ x \in B, \ 1 \leq i \leq v,$$
$$t \geq \left| \frac{\partial \mathcal{S}_i(x)}{\partial x_j} \right| \ \forall \ x \in B, \ 1 \leq i, j \leq v.$$

We call a map $\mathcal{M}$ shrinkable if $(1 - vt) > 0$ and $(1 - s) > 0$; both of which can be achieved if $\mathcal{S}$ (and since it is a polynomial, also its derivative) is sufficiently small in magnitude. Then we define q, the so-called shrink wrap factor, as

$$q = 1 + d \cdot \frac{1}{(1 - (v - 1)t) \cdot (1 - s)}.$$

The bounds s and t for the polynomials $\mathcal{S}_i$ and $\partial \mathcal{S}_i / \partial x_j$ can be computed by interval evaluation. The factor q will prove to be a factor by which the Taylor polynomial $\mathcal{I} + \mathcal{S}$ has to be multiplied in order to absorb the remainder bound interval.

Remark 1. *(Typical values for q) To put the various numbers in perspective, in the case of the verified integration of the Asteroid 1997 XF11, we typically have $d = 10^{-7}$, $s = 10^{-4}$, $t = 10^{-4}$, and thus $q \approx 1 + 10^{-7}$. It is interesting to note that the values for s and t are determined by the nonlinearity in the problem at*

hand, while in the absence of "noise" terms in the ODEs described by intervals, the value of d is determined mostly by the accuracy of the arithmetic. Rough estimates of the expected performance in quadruple precision arithmetic indicate that with an accompanying decrease in step size, if desired d can be decreased below 10^{-12}, *resulting in* $q \approx 1 + 10^{-12}$.

In order to proceed, we need some estimates relating image distances to origin distances.

Lemma 1. *Let* $\mathcal{M}$ *be a map as above, let* $\|\cdot\|$ *denote the max norm, and let* $(1-vt) > 0$. *Then we have*

$$\begin{aligned} |\mathcal{M}_i(\bar{x}) - \mathcal{M}_i(x)| &\leq \sum_j |\delta_{i,j} + t| \;\; |\bar{x}_j - x_j|\,, \\ \|\mathcal{M}(\bar{x}) - \mathcal{M}(x)\| &\leq (1+vt) \cdot \|\bar{x} - x\|\,, \;\; \textit{and} \\ \|\mathcal{M}(\bar{x}) - \mathcal{M}(x)\| &\geq (1-vt) \cdot \|\bar{x} - x\|\,. \end{aligned}$$

where $\delta_{i,j}$ *denotes the Kronecker delta.*

Proof. For the proof of the first assertion, we observe that all $(v-1)$ partials of $\partial\mathcal{M}_i/\partial x_j$ for $j \neq i$ are bounded in magnitude by t, while $\partial\mathcal{M}_i/\partial x_i$ is bounded in magnitude by $1+t$; thus the first statement follows from the intermediate value theorem. For the second assertion, we trivially observe

$$\begin{aligned} \|\mathcal{M}(\bar{x}) - \mathcal{M}(x)\| &= \max_i |\mathcal{M}_i(\bar{x}) - \mathcal{M}_i(x)| \\ &\leq \max_i \sum_j |\delta_{i,j} + t| \;\; |\bar{x}_j - x_j| \\ &\leq (1+vt) \;\; \|\bar{x} - x\|\,. \end{aligned}$$

For the proof of the third assertion, which is more involved, let k be such that $\|\bar{x} - x\| = |\bar{x}_k - x_k|$, and wlog let $\bar{x}_k - x_k > 0$. Then we have

$$\begin{aligned} \|\mathcal{M}(\bar{x}) - \mathcal{M}(x)\| &= \max_i |\mathcal{M}_i(\bar{x}) - \mathcal{M}_i(x)| \\ &\geq |\mathcal{M}_k(\bar{x}) - \mathcal{M}_k(x)| \\ &= \left| (1+c_k)(\bar{x}_k - x_k) + \sum_{j\neq k} c_j(\bar{x}_j - x_j) \right| \end{aligned} \tag{3.2}$$

for some set of c_j with $|c_j| \leq t \;\forall j = 1, ..., v$, according to the mean value theorem. Now observe that for any such set of c_j,

$$\begin{aligned} \left| \sum_{j\neq k} c_j(\bar{x}_j - x_j) \right| \leq \sum_{j\neq k} |c_j| \;\; |\bar{x}_j - x_j| &\leq \left(\sum_{j\neq k} |c_j| \right) \;\; |\bar{x}_k - x_k| \\ &\leq (v-1) \;\; t \;\; |\bar{x}_k - x_k| \\ &\leq (1-t) \;\; |\bar{x}_k - x_k| \leq (1+c_k)\,(\bar{x}_k - x_k)\,. \end{aligned}$$

Hence the left term in the right hand absolute value in (3.2) dominates the right term for any set of c_j, and we thus have

$$\begin{aligned}
&\left|(1+c_k)(\bar{x}_k - x_k) + \sum_{j\neq k} c_j(\bar{x}_j - x_j)\right| \\
&\geq (1-t)(\bar{x}_k - x_k) - \sum_{j\neq k} t \ \ |\bar{x}_j - x_j| \\
&\geq (1-t)(\bar{x}_k - x_k) - (v-1)\ t\ (\bar{x}_k - x_k) \\
&= (1-vt)(\bar{x}_k - x_k) = (1-vt)\ \ \|\bar{x} - x\|\,,
\end{aligned}$$

which completes the proof. □

Theorem 3. *(Shrink Wrapping) Let $\mathcal{M} = \mathcal{I} + \mathcal{S}(x)$, where $\mathcal{I}$ is the identity. Let $I = d \cdot [-1,1]^v$, and*

$$R = I + \bigcup_{x\in B} \mathcal{M}(x)$$

be the set sum of the interval $I = [-d,d]^v$ and the range of $\mathcal{M}$ over the original domain box B. Let q be the shrink wrap factor of $\mathcal{M}$; then we have

$$R \subset \bigcup_{x\in B} (q\mathcal{M})(x),$$

and hence multiplying $\mathcal{M}$ with the number q allows to set the remainder bound to zero.

Proof. Let $1 \leq i \leq v$ be given. We note that because $\partial\mathcal{M}_i/\partial x_i > 1 - t > 0$, $\mathcal{M}_i$ increases monotonically with x_i. Consider now the $(v-1)$ dimensional surface set $(x_1, ..., x_v)$ with $x_i = 1$ fixed. Pick a set of $x_j \in [-1,1]$, $j \neq i$. We want to study how far the set $R = I + \bigcup_{x\in B} \mathcal{M}(x)$ can extend beyond the surface in direction i at the surface point $y = \mathcal{M}(x_1, ..., x_{i-1}, 1, x_{i+1}, ..., x_v)$.

Let y_i be the i-th component of y. The i-th components of the set $y+I$ apparently extends beyond y_i by d. However, it is obvious that R can extend further than that beyond y_i. In fact, for any other $\bar{y}$ with $|\bar{y}_j - y_j| \leq d$ for $j \neq i$, there are points in $\bar{y} + I$ with all but the i-th component equal to those of y. On the other hand, any $\bar{y}$ with $|\bar{y}_j - y_j| > d$ for some $j \neq i$ can not have a point in $\bar{y} + I$ with all but the i-th component matching those of y. So at the point y_i, the set R can extend to

$$r_i(y) = d + \sup_{\{\bar{y}|\ |\bar{y}_j - y_j| \leq d\ (j\neq i)\}} \bar{y}_i.$$

We shall now find a bound for $r_i(y)$. First we observe that because of the monotonicity of $\mathcal{M}_i$, we can restrict the search to the case with $x_i = 1$. We now project to an $(v-1)$ dimensional subspace by fixing $x_i = 1$ and by removing the i-th component $\mathcal{M}_i$. We denote the resulting map by $\mathcal{M}^{(i)}$, and similarly denote all $(v-1)$ dimensional variables with the superscript "(i)".

We observe that with the function $\mathcal{M}$, also the function $\mathcal{M}^{(i)}$ is shrinkable according to the definition, with factors s and t inherited from $\mathcal{M}$. Apparently the condition on $\bar{y}$ in the definition of $r_i(y)$ entails that in the $(v-1)$ dimensional subspace, $\left\|\bar{y}^{(i)} - y^{(i)}\right\| \leq d$. Let $\bar{x}^{(i)}$ and $x^{(i)}$ be the $(v-1)$ dimensional pre-images

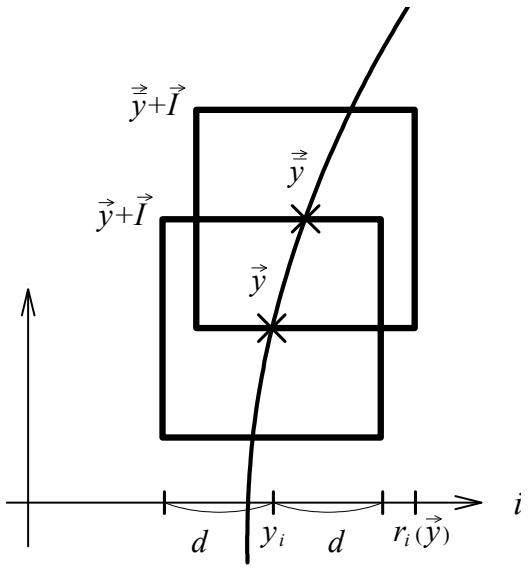

FIGURE 7. At the point y_i, the set $R = I + \bigcup_{x\in B} \mathcal{M}(x)$ can extend to $r_i(y)$.

of $\bar{y}^{(i)}$ and $y^{(i)}$, respectively; because $\left\|\bar{y}^{(i)} - y^{(i)}\right\| \le d$, we have according to the above lemma that

$$\left\|\bar{x}^{(i)} - x^{(i)}\right\| \le \frac{d}{1-(v-1)t},$$

which entails that also in the original space we have $|\bar{x}_j - x_j| \le d/(1-(v-1)t)$ for $j \neq i$. Hence we can bound $r_i(y)$ via

$$r_i(y) \le d + \sup_{\substack{\{\bar{x}|\ |\bar{x}_j - x_j| \le d/(1-(v-1)t) \\ (j\neq i),\ x_i = \bar{x}_i = 1\}}} \mathcal{M}_i(\bar{x}).$$

We now invoke the first statement of the lemma for the case of $\bar{x}$, x satisfying $|\bar{x}_j - x_j| \le d/(1-(v-1)t)$ $(j \neq i)$, $x_i = \bar{x}_i = 1$. The last condition implies that the term involving $(\delta_{i,j} + t)$ does not contribute, and we thus have $|\mathcal{M}_i(\bar{x}) - \mathcal{M}_i(x)| \le (v-1)t \cdot d/(1-(v-1)t)$, and altogether

$$\begin{aligned} r_i(y) &\le y_i + d + \frac{d\cdot(v-1)t}{1-(v-1)t} \\ &= y_i + d \cdot \frac{1}{1-(v-1)t}. \end{aligned}$$

We observe that the second term in the last expression is independent of i. Hence we have shown that the "band" around $\bigcup_{x\in B} \mathcal{M}(x)$ generated by the addition of I never extends more than $d/(1-(v-1)t)$ in any direction.

To complete the proof, we observe that because of the bound s on $\mathcal{S}$, the box $(1-s)[-1,1]^v$ lies entirely in the range of $\mathcal{M}$. Thus multiplying the map $\mathcal{M}$ with any factor $q > 1$ entails that the edges of the box $(1-s)[-1,1]^v$ move out by the amount $(1-s)(q-1)$ in all directions. Since the box is entirely inside the range of

$\mathcal{M}$, this also means that the border of the range of $\mathcal{M}$ moves out by at least the same amount in any direction i. Thus choosing q as

$$q = 1 + d \cdot \frac{1}{(1-(v-1)t) \cdot (1-s)}$$

assures that

$$\bigcup_{x \in B} (q\mathcal{M}) \supset R$$

as claimed. □

Remark 2. ***(Shrink Wrapping and Complex Arithmetic)*** *Taylor models have also been successfully used to perform operations in the complex plane. To this end, one merely identifies complex functions as functions from R^2 into R^2 and observes that analyticity entails infinite partial differentiability of the component functions. Thus complex analytic functions can be described as pairs of Taylor models in two variables, and the rules for Taylor model arithmetic can be applied to the component functions. Apparently the geometric properties of the resulting ranges of the Taylor models are analogous to the situation of the flows of ODEs above; and in a similar way it is thus possible to absorb the remainder term into the polynomial part of the Taylor model.*

Let us consider the practical limitations of the method:

Remark 3. ***(Limitations of Shrink Wrapping)*** *Apparently the shrink wrap method discussed above has the following limitations*

1. *The measures of nonlinearities s and t must not become too large*
2. *The application of the inverse of the linear part should not lead to large increases in the size of remainder bounds.*

Apparently the first requirement limits the domain size that can be covered by the Taylor model, and it will thus be relevant only in extreme cases. Furthermore, in practice the case of s and t becoming large is connected to also having accumulated a large remainder bound, since the remainder bounds are calculated from the bounds of the various orders of s. In the light of this, not much additional harm is done by removing the offending s into the remainder bound and create a linearized Taylor model.

Definition 3. ***(Linearized Taylor Model)*** *Let $M_0 \cdot x + \mathcal{S} + I$ be a Taylor model with nonlinear part $\mathcal{S}$, and let the components of S be bounded by $s = (s_i)$. We call*

$$M_0 \cdot x + I + s \cdot [-1, 1]$$

the linearized Taylor model of $M_0 \cdot x + \mathcal{S} + I$.

The overestimation generated by the application of the inverse of the linear part is apparently directly connected to the condition number of the linear part M_0.

Definition 4. ***(Blunting of an Ill-Conditioned Matrix)*** *Let A be a regular nxn matrix that is potentially ill-conditioned and $q = (q_1, ...q_n)$ be a vector with*

$q_i > 0$. Arrange the column vectors a_i of A by Euclidean length. Let e_i be the familiar orthonormal vectors obtained through the Gram-Schmidt procedure, i.e.

$$e_i = \frac{a_i - \sum_{k=1}^{i-1} e_k \ (a_i \cdot e_k)}{\left| a_i - \sum_{k=1}^{i-1} e_k \ (a_i \cdot e_k) \right|}.$$

We form vectors b_i via

$$b_i = a_i + q_i e_i$$

and assemble them columnwise into the matrix B. We call B the q-blunted matrix belonging to A.

Proposition 1. ***(Regularity of the Blunted Matrix)*** *The b_i are linearly independent and thus B is regular.*

Proof. By induction. Apparently b_1 is linearly independent. Assume now that $b_1, ..., b_{i-1}$ are linearly independent. We first observe that for each i, the vector b_i is by virtue of its definition a linear combination of the a_k for $k = 1, ..., i$ and thus also of the e_k for $k = 1, ..., i$, since both sets of vectors span the same space. Now suppose b_i is linearly dependent on $b_1, ..., b_{i-1}$; then it is also linearly dependent on $e_1, ..., e_{i-1}$, and in particular we must have $b_i \cdot e_i = 0$. Observe that we have $(a_i)^2 = \sum_{k=1}^{n} (a_i \cdot e_k)^2$ by virtue of the fact that the vectors e_k form an orthonormal basis. Using this, we obtain from the definition of b_i that

$$\begin{aligned} b_i \cdot e_i &= a_i \cdot e_i + q_i \\ &= \frac{(a_i)^2 - \sum_{k=1}^{i-1} (a_i \cdot e_k) \ (a_i \cdot e_k)}{\left| a_i - \sum_{k=1}^{i-1} e_k \ (a_i \cdot e_k) \right|} + q_i \\ &= \frac{\sum_{k=i}^{n} (a_i \cdot e_k)^{\ 2}}{\left| a_i - \sum_{k=1}^{i-1} e_k \ (a_i \cdot e_k) \right|} + q_i > 0, \end{aligned}$$

which represents a contradiction to $b_i \cdot e_i = 0$; thus $b_1, ..., b_i$ are linearly independent, which completes the induction step. □

Remark 4. ***(Effect of Blunting)*** *The intuitive effect of the blunting is that b_1, and thus the dominating direction, which determines asymptotic behavior, remains unchanged. Smaller b_i are being "pulled away" from earlier ones in the direction of e_i, i.e. away from the space spanned by the previous vectors $b_1, ..., b_{i-1}$. Since $b_i \cdot e_i \geq q_i$, the "pulling" is stronger for larger choices of q_i. Thus larger choices for q_i lead to a matrix that has more favorable condition number.*

Algorithm 1. ***(Pre-Conditioning of Shrink Wrapping)*** *Let M_0 be the linear part of the Taylor model to be shrink wrapped. Subject M_0 to the blunting algorithm just described before attempting to compute its inverse. As a result, M_0 is less ill-conditioned, its approximate inverse $\bar{M}_0^{-1}$ is determined more easily, and is itself less ill-conditioned. As discussed in the main algorithm, the defect of applying $\bar{M}_0^{-1}$*

to M_0 is moved to the remainder bound. Next, determine if the Taylor model is shrinkable as defined in 2. If it is not, or if the shrink wrap factor q exceeds a prespecified threshold $q_{\max}$, bound the nonlinear part into the remainder bound. The result is a shrinkable Taylor model.

Remark 5. ***(Shrink Wrapping for Linear Systems)*** *When applied to linear systems, the shrink wrapping with blunting limits the overestimation due to the conditioning of matrix when transforming the error interval to the new coordinate system. At the same time, the leading direction remains unchanged, and thus there is no error introduced that scales with the length of the leading direction, which determines the asymptotic error. On the other hand, the naive shrink wrapping method without blunting behaves like the well-known parallelepiped method.*

Apparently the trade-off of blunting the linear part lies in an increase in the size of the remainder bound that then has to be absorbed into the Taylor model. However, this increase is not affected by the size of the dominating vector, since it remains unaffected by the blunting algorithm. Thus in studies of asymptotic behavior where the other directions become exponentially smaller compared to the dominating direction, the effect of blunting will become exponentially less significant. Since this requires sending the remainder bound through the inverse, which produces an overestimation increasing with condition number, it is expected that a moderate amount of blunting and the corresponding decrease in condition number will overall lead to a smaller shrink wrap factor. Furthermore, we observe that the less ill-conditioned inverse that results from blunting will also lead to smaller nonlinear terms, which leads to a more favorable shrink wrap factor, or may even prevent the breakdown of shrinking and the need to absorb the nonlinearities into the remainder bound.

More specifically, the larger the size of the remainder bound relative to the size of the range into which it is to be absorbed, the larger the blunting factor should be chosen, since the more important overestimation by application of the inverse becomes, while the less important the additional contributions from packing the original matrix in the blunted matrix becomes. A large ensemble of examples for the use of shrink wrapping under blunting will be studied in the next section.

In a practical environment, one may even use trial and error or other heuristics to determine suitable blunting parameters. Also, much further theoretical thought could be spent on the question of the optimal enclosure of one parallelepiped (the remainder interval) in another (the linear part). For example, one could attempt to find a "minimal" parallelepiped to do that; part of the problem is specifying the meaning of "minimal". One could think of minimizing volume, which would lead to a constrained nonlinear optimization problem. One may also think of minimizing the lengths of the vectors, which may lead to a linear programming problem.

The trade-off between these two cases seems far from obvious; first, both cases require the choice of a coordinate system that is somehow "natural" for the system, since both volume and coordinate lengths are affected by such a choice of coordinates. Furthermore, while small volume may have obvious immediate appeal, especially in the case of nonlinear systems, it may be more desirable to operate with less "extended" objects, which may reduce subsequent nonlinear effects. Finally, if the system under consideration exhibits a particular symmetry like energy conservation or symplecticity, emphasis may be placed on the satisfaction of these

symmetries. Altogether, although of course all arguments remain validated in our setting, the efficiency of the method is greatly affected by heuristic choices, in much the same way as in conventional numerical integration.

Definition 5. ***(Parameterizing of Remainder Bounds)*** *Let (P, I) be a Taylor model describing a function $f : D \subset R^n \to R^m$ We introduce a new polynomial $P^* : (D \times I) \subset R^{n+m} \to R^m$ via*

$$P^*(x, t) = P(x) + t \text{ on } D^* = D \times I.$$

The Taylor model $(P^*, [0, 0])$ is called a parameter-extended Taylor model of f.

We have the following immediate result.

Proposition 2. ***(Enclosure Property)*** *For all $x \in D$, we have $f(x) \in P^*(x, I) + [0, 0]$*

What may appear as a simple mathematical slight of hand actually has important consequences, since for subsequent steps of the integration, we have uniquely represented f by only the Taylor model $(P^*, [0, 0])$ in a higher dimensional space that has no remainder bound. We may thus proceed with subsequent operations in Taylor model arithmetic with the parameter extended Taylor model $(P^*, [0, 0])$ instead of the Taylor model (P, I). The consequence is that in later steps, what was originally the interval I and is thus subject to the cancellation and wrapping problems, is now the variable t, which can be carried through all occurring Taylor model operations.

4. Preconditioning the Flow

In this section we will discuss another method to affect the behavior of the remainder bounds of the solutions of ODEs. The idea is to write the Taylor model of the solution as a composition of two Taylor models $(P_l + I_l)$ and $(P_r + I_r)$, and then choose $P_l + I_l$ in such a way that I_l is zero up to roundoff, and the operations appearing on I_r are minimized so as not to increase the size of I_r significantly. In a wider context, the Taylor model $(P_l + I_l)$ can be viewed as a specific coordinate system in which the motion is studied. For practical purposes, in the factorization we impose that $(P_r + I_r)$ is normalized such that each of its components has a range in $[-1, 1]$; for purposes of numerical stability, it is advantageous that the range is in fact near $[-1, 1]$.This is achieved by factoring out a linear diagonal transformation containing scaling factors.

Definition 6. *Let $(P+I)$ be a Taylor model. We say that (P_l+I_l), S, and (P_r+I_r) form a factorization of $(P+I)$ if the components of the range $B(P_r+I_r)$ of P_r+I_r lie in $[-1, 1]$, S is a diagonal linear scaling transformation, and*

$$(P + I) \in (P_l + I_l) \circ S \circ (P_r + I_r) \text{ for all } x \in D.$$

Here D is the domain of the Taylor model $(P + I)$. In this case, we call $P_l + I_l$ the preconditioner, S the scaling, and $P_r + I_r$ the conditioned Taylor model.

The composition $(P_1+I_1)\circ(P_2+I_2)$ of the Taylor models (P_1+I_1) and (P_2+I_2) is here to be understood as insertion of the Taylor model (P_2+I_2) into the polynomial P_1 via Taylor model addition and multiplication, and subsequent addition of the remainder bound I_1. For the study of the solutions of ODEs, the following result is important

Proposition 3. *Let $(P_{l,n} + I_{l,n}) \circ S_n \circ (P_{r.n} + I_{r,n})$ be a factored Taylor model that encloses the flow of the ODE at time t_n. Let $(P^*_{l,n+1}, I^*_{l,n+1})$ be the result of integrating $(P_{l,n} + I_{l,n})$ from t_n to t_{n+1}. Then*

$$(P^*_{l,n+1}, I^*_{l,n+1}) \circ S_n \circ (P_{r,n} + I_{r,n})$$

is a factorization of the flow at time t_{n+1}.

Thus the right factor remains unchanged. Considering that in the beginning of the integration, the flow of the initial condition box can be represented as the composition of two identity Taylor models, this immediately leads to the obvious but uninteresting case of leaving the right factor as the identity throughout the integration, which apparently reduces to the naive Taylor model integration. However, the key to the beneficial use of the method, and in particular its use in reducing the growth of remainder terms, lies in moving terms between the left and right factors.

To actually achieve the factorization, the following steps are necessary. First, observe that according to proposition 3, an inclusion of the flow in a Taylor model is given by $(P^*_{l,n+1} + I^*_{l,n+1}) \circ S_n \circ (P_{r.n} + I_{r,n})$. Let c^*_{n+1}, C^*_{n+1} be the constant and linear parts of $P^*_{l,n+1}$ and $N^*_{l,n+1}$ the nonlinear part and the remainder, so that $P^*_{l,n+1} = c^*_{n+1} + C^*_{n+1} + N^*_{n+1}$. We set $c_{l,n+1} = c^*_{n+1}$ and assume that $C_{l,n+1}$ is the desired linear part of the left factor; more on useful choices for $C_{l,n+1}$ below. We then insert the identity transformation $(C_{l,n+1} \circ C^{-1}_{l,n+1})$ in front of the parentheses, and thus have an inclusion of the flow as follows:

$$\begin{aligned}
&\left(c^*_{n+1} + C^*_{n+1} + N^*_{n+1}\right) \circ S_n \circ (P_{r,n} + I_{r,n}) \\
&= c^*_{n+1} + \left(C^*_{n+1} + N^*_{n+1}\right) \circ S_n \circ (P_{r,n} + I_{r,n}) \\
&= c_{l,n+1} + (C_{l,n+1} + [0,0]) \circ \left(C^{-1}_{l,n+1} \circ \left(C^*_{n+1} + N^*_{n+1}\right) \circ S_n \circ (P_{r,n} + I_{r,n})\right) \\
&= (c_{l,n+1} + C_{l,n+1} + [0,0]) \circ \\
&\left\{\left[C^{-1}_{l,n+1} \circ C^*_{n+1} + C^{-1}_{l,n+1} \circ N^*_{n+1}\right] \circ S_n \circ (P_{r,n} + I_{r,n})\right\} \qquad (4.1)
\end{aligned}$$

We now denote the expression in the curly brackets by $\left(P'_{r,n+1} + I'_{r,n+1}\right)$ and determine its component bounds, which produces the scaling matrix S_{n+1}. Denoting $(P_{r,n+1} + I_{r,n+1}) = S^{-1}_{n+1} \circ \left(P'_{r,n+1} + I'_{r,n+1}\right)$, we thus have an enclosure of the flow at t_{n+1} as

$$(c_{l,n+1} + C_{l,n+1} + [0,0]) \circ S_{n+1} \circ (P_{r,n+1} + I_{r,n+1}).$$

To analyze the effects of this procedure, the following observations are crucial:

(1) The polynomial part of $C^{-1}_{l,n+1} \circ N^*_{n+1}$ is purely nonlinear, so its action on $S_n \circ (P_{r,n} + I_{r,n})$ via composition only introduces small contributions to the remainder bound which scale at least quadratically with the components of S_{n+1}. Thus for sufficiently small S_{n+1}, this effect will be small.
(2) The remainder part of $C^{-1}_{l,n+1} \circ N^*_{n+1}$, which contains as one important contribution the action of $C^{-1}_{l,n+1}$ on the remainder interval of N^*_{n+1}, will be

added to $I_{r,n}$. The magnification of the remainder bound of N^*_{n+1} by the action of $C^{-1}_{l,n+1}$ is proportional to the condition number of $C_{l,n+1}$.

(3) Contributions of a similar magnitude as $I_{r,n}$ come from application of the linear term $C^{-1}_{l,n+1} \circ C^*_{n+1}$ to $I_{r,n}$. If this term is not chosen properly, over time, exponential growth of the remainder bound can occur.

We now are ready to consider several choices for the determination of $C_{l,n+1}$. As a first nearly trivial but nevertheless interesting example, we assume that the polynomial $P_{l,n}$ represents the identity:

Definition 7. ***(Identity Preconditioning)*** *We choose $C_{l,n+1}$ as the identity:*

$$C_{l,n+1} = \mathcal{I}$$

This form of preconditioning amounts merely to moving the remainder error to the right. In the subsequent step, the flow is then computed on an identity without the presence of a remainder bound, which can lead to improved performance. This is somewhat reminiscent of the common distinction between "algorithm 1" and "algorithm 2" of integration approaches such as those in the code AWA, where "algorithm 1" provides a first enclosure over an interval box enclosing the current flow.

As the first nontrivial but nevertheless quite obvious example, we assume that the polynomial $P_{l,n}$ represents the linear flow of the motion.

Definition 8. ***(Parallelepiped Preconditioning)*** *We choose*

$$C_{l,n+1} = C^*_{n+1}$$

The parallelepiped preconditioning thus has the interesting effect that the entire constant and linear parts of the flow are described by the left factor alone; and the nonlinear parts of the motion and remainder bounds will by accumulated in the right factor. Analyzing the arithmetic more carefully we see that the term $C^{-1}_{l,n+1} \circ C^*_{n+1} + C^{-1}_{l,n+1} \circ N^*_{n+1}$ appearing in the square brackets in eq. 4.1 plays a crucial role. Its linear part amounts to identity up to floating point error which leads to very favorable numerics in the subsequent composition with $S_n \circ (P_{r,n} + I_{r,n})$.

On the other hand, $C^{-1}_{l,n+1}$ is also acting on the nonlinear part and the remainder bound. However, it is known that in various practical cases of interest, over long periods of time, $C_{l,n+1}$ can become more and more ill-conditioned; this is for example the case in linear problems where the matrix of the ODE has distinct real eigenvalues. Since the multiplication of a matrix with an interval vector leads to an overestimation that scales with the condition number, this effect may lead to a rapid growth of the remainder bound of the term, and thus in cases of ill-conditioned flow is of limited value.

The method can be much improved by the following choice of preconditioner:

Definition 9. ***(Blunted Parallelepiped Preconditioning)*** *We choose $C_{l,n+1}$ to be the $q-$blunting of C^*_{n+1}, where q is a suitable blunting factor.*

As seen above, the q-blunting provides an upper bound for the condition number of the matrix $C_{l,n+1}$, and thus a strict upper limit to the overestimation obtained when sending the remainder bound interval of N^*_{n+1} through $C^{-1}_{l,n+1} \circ \left(C^*_{n+1} + N^*_{n+1}\right)$. On the other hand, since a sufficiently small choice of q only modifies $C_{l,n+1}$ in a minor amount, we still have that the linear part of $C^{-1}_{l,n+1} \circ$

$\left(C_{n+1}^* + N_{n+1}^*\right)$ is nearly identity, which still favorably affects the subsequent application to $S_n \circ (P_{r,n} + I_{r,n})$. So a suitable choice of q may lead to an acceptable overestimation due to the condition number of $C_{l,n+1}$ while still providing only limited overestimation in the last step. Examples of the effect of blunted parallelepiped preconditioning will be given in the next section.

As another example of preconditioning with a linear transformation, we consider the following choice

Definition 10. ***(Curvilinear Preconditioning)*** *Let $x^{(m)} = f(x, x', ...x^{(m-1)}, t)$ be an m-th order ODE in n variables. Let $x_r(t)$ be a solution of the ODE and $x_r'(t), ..., x_r^{(k)}(t)$ its first k time derivatives. Let $e_1(t), ..., e_l(t)$ be l unit vectors not in the span of $x_r'(t)$, ..., $x_r^{(k)}(t)$ such that $X = (x_r'(t), ..., x_r^{(k)}(t), e_1(t), ..., e_l(t))$ have maximal rank. Then we call the Gram-Schmidt orthonormalization of X a curvilinear basis of depth k, and we refer to its use for preconditioning as curvilinear preconditioning.*

The use of curvilinear coordinates has a long history, and it seems like their virtues have been re-discovered several times. They are frequently used in the study of dynamics in the solar system, and for the last 50 years in the dynamics in large particle accelerators. For a treatment of their properties in the latter case, see [3], and [26] as well as [29]. As an aside, we note that it is possible to even preserve Hamiltonian structure in the transformation to curvilinear coordinates[26][7], which is important for long term integration using symplectic methods as in [11] and [12].

Example 1. ***(Curvilinear Coordinates for the Solar System and Particle Accelerators)*** *In this case, $n = 3$, and one usually chooses $k = 2$. The first basis vector points in the direction of motion of the reference orbit. The second vector is perpendicular to it and points approximately to the sun or the center of the accelerator. The third vector is chosen perpendicular to the plane of the previous two.*

Theorem 4. ***(Curvilinear Coordinates for Autonomous Linear Systems)*** *Let $x' = A \cdot x$ be an n-dimensional linear system that has n distinct nonzero eigenvalues λ_i with eigenvectors a_i. Let B be a box with nonzero volume, and $x_r = \sum_{i=1}^n X_i a_i \in B$ such that $X_i \neq 0$ for all $i = 1, ..., n$. Then the derivatives of $x_r^{(i)}$, $i = 1, ..., n$, are linearly independent, and hence the depth n curvilinear coordinates are obtained by applying the Gram-Schmidt procedure to the derivatives $x_r^{(i)}$, $i = 1, ..., n$.*

Proof. The motion of the reference point x_r as a function of time is apparently given by

$$x_r(t) = \sum_{i=1}^{n} X_i \cdot a_i \cdot \exp(\lambda_i t)$$

so that the jth derivative assumes the form

$$x_r^{(j)}(t) = \sum_{i=1}^{n} X_i \cdot a_i \cdot \lambda_i^j \exp(\lambda_i t).$$

We now consider the determinant of the matrix of coefficients in the basis a_i, and observe

$$\det \begin{pmatrix} X_1\lambda_1 & X_1\lambda_1^2 & & X_1\lambda_1^n \\ X_2\lambda_2 & X_2\lambda_2^2 & & X_2\lambda_2^n \\ & & \ddots & \\ X_n\lambda_n & X_n\lambda_n^2 & & X_n\lambda_n^n \end{pmatrix}$$
$$= \prod_{i=1}^{n} (\lambda_i X_i) \cdot \det \begin{pmatrix} 1 & \lambda_1^1 & & \lambda_1^{n-1} \\ 1 & \lambda_2^1 & & \lambda_2^{n-1} \\ & & \ddots & \\ 1 & \lambda_n^1 & & \lambda_n^{n-1} \end{pmatrix} = \prod_{i=1}^{n} (\lambda_i X_i) \prod_{i>j} (\lambda_i - \lambda_j) \neq 0$$

because of the well-known property of the Vandermonde matrix. □

Definition 11. ***(Natural Coordinate System for Linear System)*** *Let $x' = A \cdot x$ be an n-dimensional linear system that has n distinct real eigenvalues $\lambda_1 > \lambda_2 > ... > \lambda_n$ with eigenvectors $a_1, ..., a_n$. We define the normal basis (b_i) of the system to be the result of applying the Gram-Schmidt orthonormalization procedure to the vectors $a_1, ..., a_n$, i.e. the result of the recursive computation*

$$b_i = \frac{a_i - \sum_{j=1}^{i-1} b_j \cdot (a_i \cdot b_j)}{\left| a_i - \sum_{j=1}^{i-1} b_j \cdot (a_i \cdot b_j) \right|}.$$

The Natural Coordinate System has the property that as time progresses, the components motion is pulled most towards the vectors b_1, and then towards b_2, and so on.

Proposition 4. ***(Curvilinear Coordinates for Autonomous Linear Systems)*** *Let $x' = A \cdot x$ be an n-dimensional linear system that has n distinct real eigenvalues λ_i with eigenvectors a_i. Let b_i be the natural coordinate system of the linear system. Let B be a box with nonzero volume, and $x_r = \sum_{i=1}^{n} X_i a_i \in B$ such that $X_i \neq 0$. If x_r is used as the reference orbit to define the curvilinear coordinates c_i, then the curvilinear coordinates converge to the natural coordinates, i.e. we have*

$$c_i \to b_i \text{ for all } i \text{ as } t \to \infty.$$

Proof. The derivatives of the motion of the reference point x_r as a function of time of order 0 and higher are apparently given by

$$x_r^{(j)}(t) = \sum_{i=1}^{n} X_i \cdot a_i \cdot \lambda_i^j \exp(\lambda_i t)..$$

Because of the ordering of the eigenvectors by size, we clearly have $c_1 = x_r'(t)/|x_r'(t)| \to b_1$ as $t \to \infty$. Since c_2 is perpendicular to c_1, we thus also have that $c_2 \cdot b_1 \to 0$ as $t \to \infty$, and so $\lim_{t\to\infty} c_2$ is in the span of $b_2, ..., b_n$. Because in this subspace, the coefficient $\exp(\lambda_2 t)$ is dominating, we even have $c_2 \to b_2$ as $t \to \infty$. In a similar fashion we obtain iteratively that $c_j \to b_j$ as $t \to \infty$. □

Remark 6. *Variations of these arguments are obviously possible to treat the case of complex eigenvalues. In this case, the "natural" generalization of the natural coordinate system has two non-uniquely defined vectors in the subspace belonging to the conjugate pair of eigenvalues.*

Remark 7. ***(Depth of Curvilinear Coordinates)*** *One may wonder about the significance of the depth of curvilinear coordinates chosen, i.e. the number of derivatives employed. As long as the first k eigenvectors are of larger magnitude than the subsequent ones, then the subspace spanned by the first k derivatives will be asymptotically dominating over the remaining subspace, and thus the detailed choices of subsequent basis elements are insignificant as long as the basis matrix remains well-conditioned.*

Definition 12. ***(QR Preconditioning)*** *We choose $C_{l,n+1}$ to be the matrix Q of the QR factorization of the matrix obtained by sorting the columns of C^*_{n+1} by size in descending order.*

So the matrix $C_{l,n+1}$ is chosen in the same fashion as originally proposed by Lohner [21][22][20][23][24][25]. Different from his algorithm, also the Taylor model describing the linear and nonlinear parts of the motion is expressed in this coordinate system. This entails that the coefficients of this polynomial are subjected to smaller coordinate transformations, which leads to reduced roundoff errors. And of course, the transformations relating initial and final conditions are not merely linear, but nonlinear.

Like the curvilinear preconditioning method, the QR preconditioning leads to a coordinate system that is orthogonal, and thus the transformation in and out of this system is computationally benign because of the favorable condition number of the system. However, there are more similarities between curvilinear preconditioning and QR preconditioning:

Proposition 5. ***(QR Coordinates for Autonomous Linear Systems)*** *Let $x' = A \cdot x$ be an n-dimensional linear system that has n distinct nonzero eigenvalues λ_i with eigenvectors a_i. Let b_i be the natural coordinate system of the linear system and c_i the basis vectors of the QR coordinate system. Then we have*

$$c_i \to b_i \text{ for all } i \text{ as } t \to \infty.$$

The proof follows from the arguments developed in the work of Nedialkov and Jackson [38]. As a consequence, we obtain that for the important case of linear autonomous systems, the asymptotic behavior of the QR method and the curvilinear method are identical.

To illustrate the performance of the curvilinear (CV) and QR preconditioning, both of which provide orthogonal coordinate systems in which the motion is studied, let us consider the example of the simple linear ODE $x_1' = x_1$, $x_2' = x_1$. It has distinct eigenvalues ± 1 , and the eigenvector belonging to the larger eigenvalue $+1$ is $(1, 1)$, thus asymptotically, the motion is "pulled" towards this eigenvector. Figure 8 shows that in the CV preconditioning, one of the coordinate axes is attached to the direction of motion, and thus the axis will eventually line up with the vector $(1, 1)$. In the case of the QR preconditioning, where one of the vectors is always attached to the longer domain box, the motion is less regular but leads to the same asymptotic behavior, since eventually also the direction of main elongation of the solution set aligns itself with the direction of motion.

For the purpose of a nonlinear example, we use the Volterra ODE and initial conditions 2.1 from above. Figure 9 shows the coordinate systems for the case of

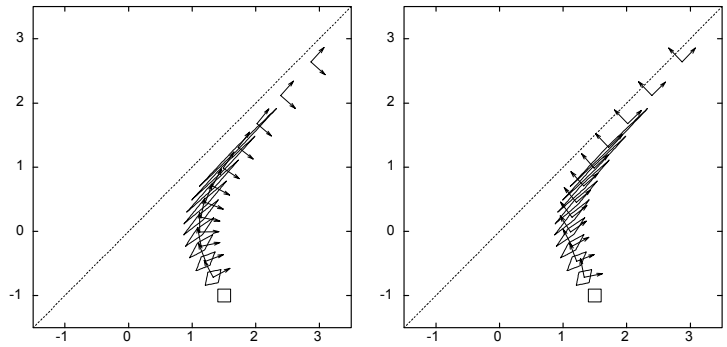

FIGURE 8. Preconditioning coordinate systems for the ODE $x_1' = x_2$, $x_2' = x_1$. Left: curvilinear, right: QR

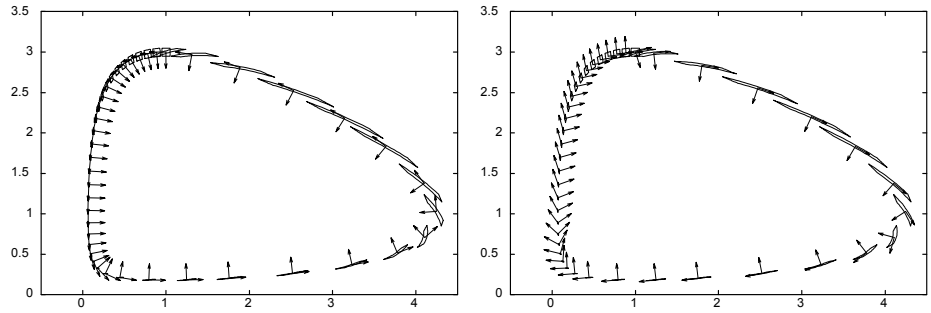

FIGURE 9. Preconditioning coordinate systems for the Volterra equations. Left: curvilinear method, right: QR method

the curvilinear preconditioning (left) and the QR preconditioning. The curvilinear coordinate system performs a full rotation by 2π upon return to the initial condition by virtue of the fact that after one full period, necessarily also the direction of the tangent to the orbit is reproduced exactly. The coordinate system used by the QR method is less regular, and it can be seen that after one revolution of the center point, the coordinate system is not rotated by 2π. The long-term success of the QR method rests on the ability to asymptotically produce rotations by 2π for each revolution of the reference point, since any persistent lag in angle will produce linear wrapping. In nonlinear systems, it is not a priori clear that this condition must always be satisfied.

5. Examples for Long-Term Behavior

The long-term numerical study of differential equations and dynamical systems in a computer environment operating with fixed precision is frequently characterized by an exponential growth of the error. We first observe the important point that this fact is intimately tied to the use of arithmetic of finite precision, and does not merely appear in validated methods. We also observe that this effect is independent of the well-known and frequently studied phenomenon of chaos, which is characterized by exponential growth of errors in initial conditions in the true system.

To illustrate this phenomenon, let us consider the perhaps simplest conceivable discrete dynamical system, which merely oscillates between two states as

$$x_{n+1} = \begin{cases} a \cdot x_n, & n \text{ even} \\ (1/a) \cdot x_n, & n \text{ odd} \end{cases} \tag{5.1}$$

with initial condition $x_0 = 1$. We study the behavior for specific choices of a in both single and double precision arithmetic on two commonly used compilers, the f77 compiler by DEC, which is now distributed as f77 Digital Visual Fortran Version 5.0 as part of Microsoft Fortran PowerStation, as well as the g77 compiler distributed by GNU; we specifically tested Version V0.5.24. All tests were executed in the Cygwin Unix environment in Windows 2000 and run on a Pentium III processor, and no changes to default rounding modes were made.

Specifically, we chose $a_1 = 3$ in the single precision mode, while in the double precision mode we chose $a_2 = 0.9999999901608054$ (digits generated by FORTRAN output)

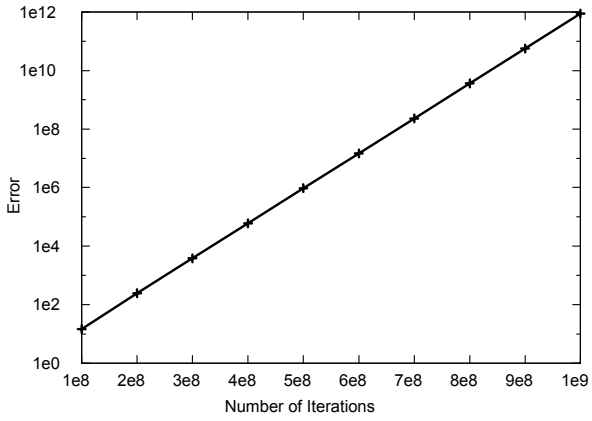

FIGURE 10. Arithmetic error observed in the computation of $x_{n+1} = (1/3) \cdot x_n$, $x_{n+2} = 3 \cdot x_{n+1}$, with $x_1 = 1$, for various values of n.

Figure 10 shows the result for the case of single precision computation using f77 with default compiler settings, revealing an exponential growth of the error that after merely 10^9 iterations reaches the value of 10^{12}. The error growth per iteration corresponds to approximately $1 + 1.2 \cdot 10^{-8}$, and hence represents an average increment near the last significant bit.

Performing the same experiment with $a_1 = 3$ in double precision arithmetic on either f77 or g77 did not produce any exponential growth of errors; however,

performing a random search for values of a near 1 that might lead to exponential growth yielded the above a_2 within the first 10 tries, and many other values of a with a similar behavior have also been found quite easily. The empirically computed error growth factor per iteration is about $1 + 1.1 \cdot 10^{-16}$, again corresponding to an increment near the last significant bit.

Executing the simple dynamical system with interval arithmetic leads to exponentially inflating bounds, as is expected from interval methods; however, in a well-written interval environment that rounds by a minimally sufficient amount, the overestimation of the computed bounds tightly enclose the growing error. Thus in this case, the observed exponential growth of the interval results is not due to any artificial overinflation of the interval method, but rather to the unavoidable uncertainty of the results of the underlying floating point arithmetic.

5.1. **Nonlinear Problems and Shrink Wrapping.** Let us now study such two-state systems in the multidimensional nonlinear setting. First we observe that any errors that may occur lead to a more complicated geometric shape of the solution sets that have to be studied. While in the one-dimensional case, an interval can always tightly contain the results of all such overestimations, this no longer holds in the multidimensional case. As a simple example, consider the following two-state discrete dynamical system

$$\begin{aligned} x_{n+1} &= x_n \cdot \sqrt{1 + x_n^2 + y_n^2} \text{ and } y_{n+1} = y_n \cdot \sqrt{1 + x_n^2 + y_n^2} \\ x_{n+2} &= x_{n+1} \cdot \sqrt{\frac{2}{1 + \sqrt{1 + 4(x_{n+1}^2 + y_{n+1}^2)}}} \text{ and} \\ y_{n+2} &= y_{n+1} \cdot \sqrt{\frac{2}{1 + \sqrt{1 + 4(x_{n+1}^2 + y_{n+1}^2)}}}. \end{aligned} \tag{5.2}$$

Simple arithmetic shows that, similar to the two-state system in eq. 5.1, also this transformation has the property that $(x_{n+2}, y_{n+2}) = (x_n, y_n)$. Considering the action of the system on the box $[-d, d]^2$, we see that the corner points $(\pm d, \pm d)$ are stretched out more than the axis intersection points $(\pm d, 0)$ and $(0, \pm d)$, which leads to a pincushion shape with four-fold symmetry after each odd step; the action on three centered squares is shown in figure 11. Attempting to represent this structure by an interval box, or for that matter any linear transformation of an interval box, will thus necessarily lead to a noticeable overestimation. On the other hand, representing the action of the iteration by a Taylor model will, for moderate values of d, be able to lead to a much more accurate representation. Finally, note that the linear transformations of the action of this system will always return to the identity after even numbers of iteration and is also rather well conditioned after odd iterations, so numerical difficulties due to conditioning do not arise in this case. Thus the example represents a good test for a method to treat nonlinear effects.

The results of a simulation with Taylor models of various orders and with and without shrink wrapping are shown in figures 12 for the point $(0, 0) + [.05, .05]^2$ and in figure 13 for the point $(1, 1) + [.05, .05]^2$. Because after two steps the linear part is the identity, the problem allows to study the ability of the shrink wrap method to

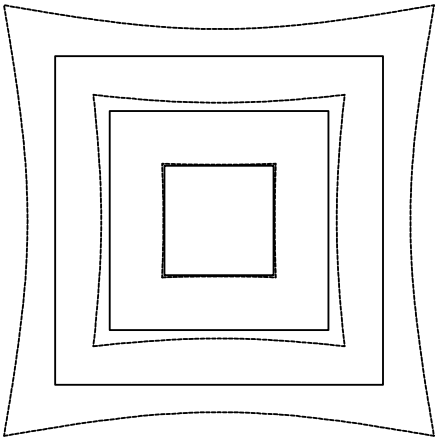

FIGURE 11. The action of the two-step nonlinear transformation. Squares are subjected to pincushion-shaped deformation and transformed back into themselves.

handle nonlinear effects, but without possible complications that may arise due to the conditioning of the linear part, which will be studied in other examples below.

Because the linear part represents the identity, shrink wrapping with first order Taylor model behaves exactly like the QR and PE methods, and so a useful comparison to these methods is possible. Apparently the use of shrink wrapping and higher order Taylor models leads to very extended stability; for example, Taylor models of order 20 lead to survival for 10^5 iterations with an accumulated error around 10^{-9}, while the lack of use of shrink wrapping or the use of linear methods leads to unacceptable errors in 100 or less iterations.

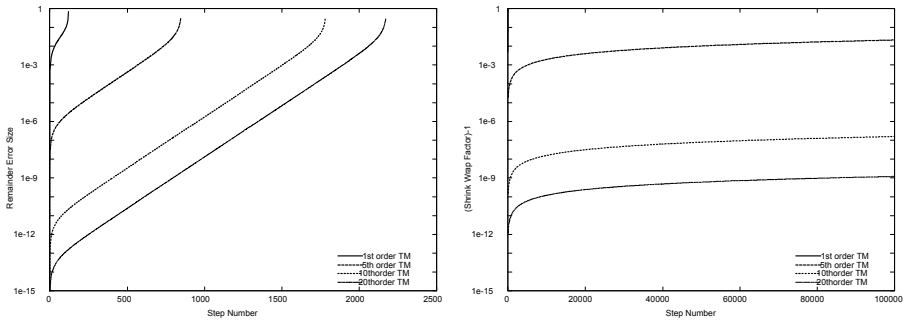

FIGURE 12. Discrete dynamics of the nonlinear stretch at $(0,0)+[-.05,.05]^2$. Treatment by naive Taylor models (left) and Taylor models with shrink wrapping (right). First order Taylor models without shrink wrapping behave like the linear PE, QR, or PEQR methods.

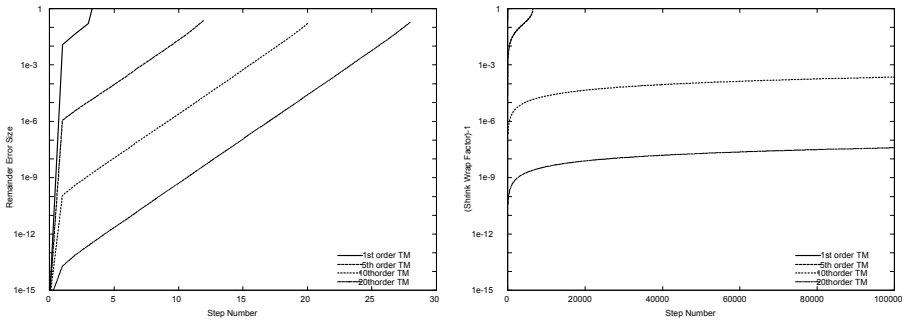

FIGURE 13. Discrete Dynamics of the nonlinear stretch at $(1,1)+[-.05,.05]^2$. Treatment by naive Taylor models (left) and Taylor models with shrink wrapping (right). First order Taylor models without shrink wrapping behave like the linear PE, QR, or PEQR methods.

5.2. **Linear Problems and Preconditioning.** While in the previous section, the emphasis was on the treatment of nonlinear effects in the absence of complications due to linear conditioning, in this section we will study the opposite: we will address linear problems that may become ill-conditioned and forgo the study of nonlinear effects. Because linear problems lead to a merely linear dependence on initial conditions, they thus allow a clear separation of the effects of the Taylor model methods that are due to the expansion in initial conditions and those of their asymptotic behavior. We consider both autonomous problems, the asymptotic behavior of which can apparently also be studied more efficiently with validated eigenvalue/eigenvector tools, as well as a specific case of a non-autonomous problems. Both of these cases allow to devise certain challenges for validated integrators, and thus represent a *sine qua non*.

We begin the analysis of the behavior of the various methods by studying discrete dynamics of iteration through two-dimensional matrices. To minimize the influence of particular choice, we consider a collection of 1000 matrices with coefficients randomly chosen in the interval $[-1,1]$. The initial condition under study is chosen to be $(1,1)+d\cdot[-1,1]$ with a value of $d=10^{-3}$. Apparently the choice of the center point of the domain box is rather immaterial due to the randomness of the matrices; and because of linearity, the value of d is of importance only relative to the floor of precision of the floating precision environment.

In all cases, we study the development of the area of enclosure as a measure of the sharpness of the method. We compare preconditioning the Taylor models by the blunted method (TMB), the parallelepiped method (TMP), and the QR method (TMQ). In this linear scenario, the TMB method also describes the effects of the blunted shrink wrapping method, which in this case also reduces to sending the remainder term through the blunted linear matrix. We chose the blunting factors q_i to be 10^{-3} times the length of the longest column vector of the linear matrix. In order to provide a frame of reference, we also study the performance of naive interval (IN) method as well as the naive Taylor model method (TMN); in the latter case, the area is estimated as the sum of the determinant of the linear part plus

the area of the remainder bound interval box. In addition, in order to provide an assessment of the influence of the effects of the underlying floating point arithmetic, we also perform a non-validated tracking of the vectors of the four corner points $(1,1) + d \cdot (\pm 1, \pm 1)$ and determine the area of the linear structure spanned by the vectors; this method is referred to as the vector method (VE). Since this method is naturally inaccurate in particular for strongly elongated structures, we average over a large number of matrices to control statistical fluctuations.

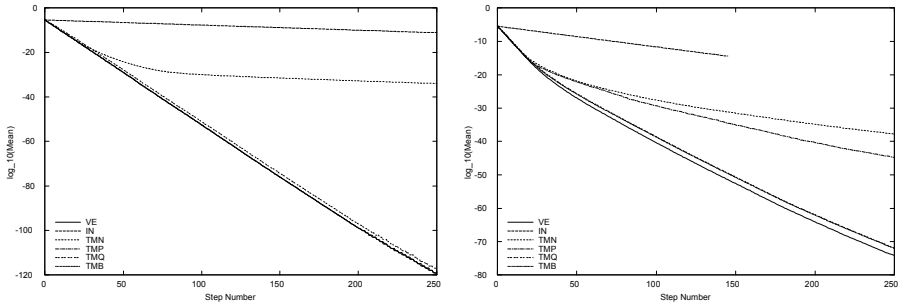

FIGURE 14. Areas predicted in the iteration through random 2x2 matrices with conjugate eigenvalues (left) and eigenvalues differing in magnitude by a factor of 1 to 5 for various enclosure methods

In the first test, we study an autonomous problem for 500 iterations. Apparently in this case, the true solution of the problem shows an exponential shrinkage of the area by the product $|\lambda_1| \cdot |\lambda_2|$ of the magnitudes of the eigenvalues. For the purpose of analysis, we group the matrices in six categories; the category C_1 contains all matrices in which the eigenvalues form conjugate pairs. The other matrices are sorted into categories based on the ratios $r = |\lambda_1|/|\lambda_2|$ of the eigenvalue λ_1 of larger magnitude to the one of smaller magnitude. Specifically we consider the categories C_2 with $1 \le r < 5$, C_3 with $5 \le r < 10$, C_4 with $10 \le r < 20$, C_5 with $20 \le r < 50$, and C_6 with $50 \le r$. The numbers of matrices in categories C_1 through C_6 are 325, 520, 80, 40, 18, and 17. Within each category, we calculate the average of the logarithm of the areas enclosed by the various methods as a function of the iteration number, which for the true dynamics would lead to a decrease along a straight line, the slope of which is given by the value $\log(|\lambda_1| \cdot |\lambda_2|)$.

Figure 14 shows the results of the situation for categories C_1 and C_2. It is clearly visible that in the dynamics of C_1, the behavior is characterized by the expected linear decrease, and the blunted (TMB), parallelepiped (TMP), and QR method (TMQ) all show this behavior. All three of these methods very closely follow the non-validated result (VE), with a closer inspection showing that the TMB and TMP methods provide enclosures about 1 to 2 orders of magnitude sharper than the TMQ method. The behavior of the methods is in agreement with the theoretical results and practical examples found in [38]. On the other hand, the naive interval method (IN) as well as the naive Taylor model method (TMN) show a qualitatively different behavior; the interval method leads to a different slope, while over the short term the naive Taylor model method performs similar to the other methods until the size of the remainder bound becomes the dominating contribution, at which time its slope becomes similar to that of the interval method.

Studying the behavior of the class C_2 shows a similar pattern, except that now the TMB and TMQ methods provide indistinguishable sharpness, while the parallelepiped method now performs markedly worse. This is due to the unfavorable conditioning of the TMP approach that does not appear in the TMQ approach.

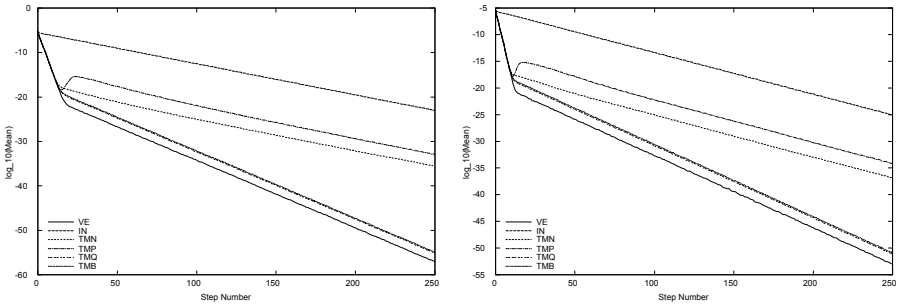

FIGURE 15. Areas predicted in the iteration through random $2x2$ matrices with eigenvalues differing in magnitude by a factor between 5 and 10 (left) and 10 and 20 (right) for various enclosure methods

Studying the behavior in the classes C_3 and C_4 shown in figure 15 reveals again that the TMB and TMQ methods perform virtually indistinguishable, and both of them follow the non-validated result VE very closely. Furthermore, the naive Taylor model method TMN and the parallelepiped method TMP both perform quite similar to each other, but substantially worse than the TMB and TMQ methods.

However, another interesting effect appears. We notice that there is a marked change in the slope of the curve after somewhere around $n = 20$ iterations for the C_3 case and $n = 15$ iterations for the C_4 case. This is attributed to the fact that after this number of iterations, the quantity r^n reaches around 10^{17}, and thus the ratio of the elongations of the solution domain in the directions of the eigenvectors v_1 and v_2 reaches the limit of what can be represented in a double precision floating point environment. Before this value of n, the computed volume decreased by $\lambda_1 \cdot \lambda_2$ at each iteration, but after this n, the apparent "thickness" of the needle-like structure will be determined by the floating point accuracy ε times the length of the needle. Thus any decrease in volume is merely due to the decrease of the needle's length, which is governed by the eigenvalue of smaller magnitude λ_2, and so the subsequent volume decrease is given by λ_2^2. Thus one is bound to observe a jump in slope of about the factor r.

Thus we observe that in the process of floating point errors, the long-term behavior of the area is predicted qualitatively wrong, and thus does not follow the predictions of [38] for the infinite precision case anymore. However, it is most noticeable that this effect does not only appear within the validated setting, but just in the same way in the non-validated case. In the latter case, the perceived "thickness" of the needle is merely given by floating point rounding errors that prevent the four corner points from being collinear, where again the deviation from collinearity being given by the ε times the length of the respective vectors, which leads to a perceived area very similar to that in the validated case. This observation appears most important, since it stresses that the spurious exponential growth

observed compared to the true result is an unavoidable consequence of the floating point environment per se and has nothing to do with the attempt to do validated computation.

The situation for the cases of C_5 and C_6 are similar to those of the C_3 and C_4 cases, except that as expected the change in slopes appears earlier and is more pronounced.

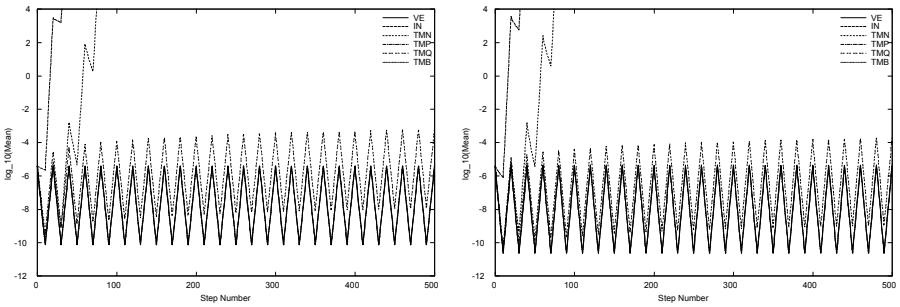

FIGURE 16. Predicted areas for 10 forward and 10 backward iterations through random $2x2$ matrices with conjugate eigenvalues (left) and eigenvalues differing in magnitude by a factor of 1 to 5 for various enclosure methods

As another set of test cases, we want to perform a limited study of non-autonomous linear systems. This case is interesting because the quantitative analysis of the behavior of the QR methods undertaken in [38] does not hold in this case, and as already observed by Kühn, spurious exponential error growth is possible; thus a comparison to the TMB method is worthwhile.

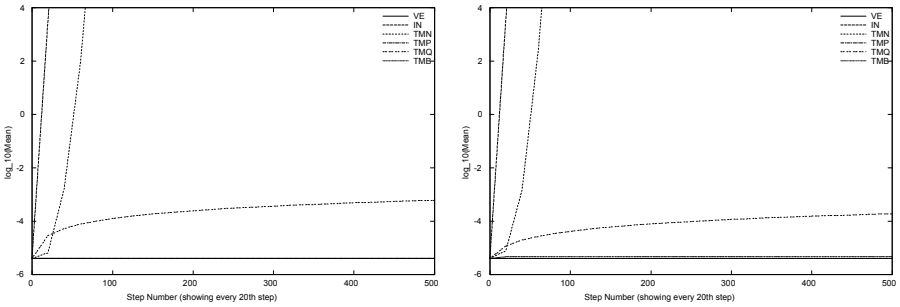

FIGURE 17. Predicted areas for groups of 10 forward and 10 backward iterations through random 2x2 matrices with conjugate eigenvalues (left) and eigenvalues differing in magnitude by a factor between 1 and 5 (right) for various enclosure methods. Results shown after each set of 20 steps.

For the purposes of our non-autonomous study, we merely iterate through the 1000 random matrices for 10 iterations, and follow these by iterating through the approximate floating point inverses of the respective matrices for the next 10 iterations, repeating this procedure a total of 25 times. So similar to the examples 5.1 and 5.2 in the previous section, the overall transformation reaches the identity after each 20 steps, and thus the analysis of the performance is straightforward. Figure 16 shows the behavior of the various methods for the case of conjugate eigenvalues and the case $1 \leq q < 5$; it is clearly seen that the naive interval (IN) and the naive Taylor model (TMN) methods lead to overestimation rather quickly. For the purpose of better readability, in figure 17 we show the enclosure area only after every 20 steps, at which point the overall transformation reaches identity. It can be seen that the TMB (and the TMP) methods reproduce the correct result to printer resolution, while the TMQ method reaches an overestimation of two orders of magnitude. For the case $1 \leq r < 5$, which is more favorable to the QR approach, again the TMB (and the TMP) method produce very little overestimation, while the TMQ method has about one order of magnitude of overestimation. For larger values of r, the advantage of the TMB method becomes less pronounced but is still one order of magnitude, while the TMP method begins to produce larger overestimations, as can be seen in figures 18 for the cases of $5 \leq r < 10$ and $10 \leq r < 20$.

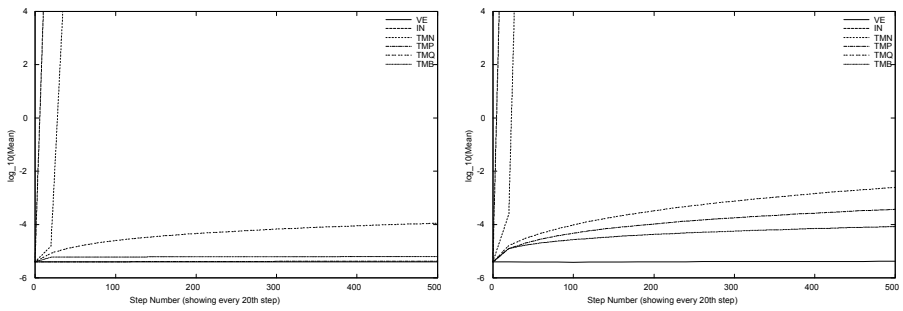

FIGURE 18. Predicted areas for groups of 10 forward and 10 backward iterations through random 2x2 matrices with eigenvalues differing in magnitude by a factor between 5 and 10 (left) and 20 and 50 (right) for various enclosure methods. Results shown after each set of 20 steps. The blunted method (TMB) outperforms the QR method.

As another example for the use of preconditioning tools for linear problems, we study some continuous problems and compare the behavior of the QR preconditioning method with the curvilinear (CV) preconditioning methods. We study an ensemble of 4x4 matrices with random elements in $[-1, 1]$, and determine validated solutions of the linear homogeneous ODE

$$r' = A \cdot r$$

over the time domain $[0, 10]$ for initial domain box $r + [-.1, .1]^4$ where r is a vector with random number entries.

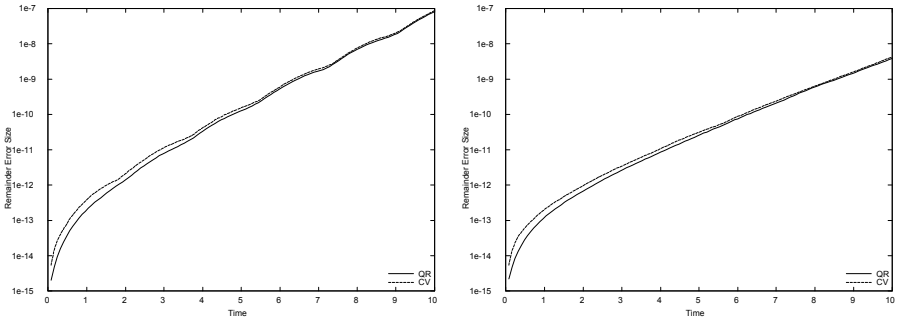

FIGURE 19. The size of the interval remainder errors for 4x4 linear systems as determined using QR and CV preconditioning. Left: averages of the errors of the four components for the matrix A_1, right: averages of the errors in the four components of 10 random matrices

We study the details of the situation for one particular matrix A_1 with approximate form

$$A_1 = \begin{pmatrix} +0.9564 & +0.2004 & +0.4826 & +0.8871 \\ -0.4922 & +0.5651 & -0.1474 & -0.7678 \\ -0.0269 & -0.8587 & -0.3785 & -0.6168 \\ -0.8271 & +0.2661 & -0.9380 & +0.5289 \end{pmatrix}$$

and approximate eigenvalues 0.3928, -0.3911, $1.005 \pm 0.8669i$ as well as for a set of 10 random matrices. The matrix A_1 was selected because it has positive, negative, and complex conjugate eigenvalues, and the complex conjugate pair is even dominating in magnitude. The random center point of the initial domain box was approximately $(0.6446, 0, 0050, -0.2394, 0.4581)$. The other matrices were studied to give confidence that what is observed is not an isolated case.

The left picture in figure 19 shows the effects of QR and CV preconditioning for the specific case A_1. Note that both remainder estimates are increasing exponentially, which is due to the fact that the magnitude of the leading eigenvalues, those that form the complex conjugate pair, exceeds unity. Apparently the two methods behave very similarly, where in the very beginning the QR preconditioning provides results that are sharper by about a factor of 2. Note that there is an oscillatory pattern visible, which is due to the fact that two of the four eigenvalues of the matrix form a complex conjugate pair, resulting in some oscillatory motion in one of the invariant subspaces of the matrix.

An attempt of a quantitative analysis of the figure shows that after the initial period of rapid error growth, which is due to the proximity of the floating point accuracy floor, the function rises exponentially from 10^{-11} at $t = 3$ to 10^{-7} at $t = 10$, which corresponds to a gain of $10^{4/7} \approx 10^{.5715}$ per time unit. On the other hand, the magnitude of the complex eigenvalue is approximately 1.327, leading to a gain of $\exp(1.327) \approx 3.769 \approx 10^{0.5763}$ per time unit. So we see that to very good approximation, the growth in the remainder error matches the growth of the parallelogram enclosing the flow of the initial domain box or the corner points thereof, which is the behavior observed in a non-validated integrator.

To conclude the discussion of linear the study of linear problems with preconditioning, we summarize the observed behavior of the methods:

(1) For iteration through identical matrices, which corresponds to study of autonomous systems, the Blunted Method and the QR method have the same asymptotic behavior and error growth as the non-validated method. On the other hand, the naive interval method, the parallelepiped method, and the naive Taylor model produce overestimations that grow exponentially.
(2) For iteration through sets of matrices and their inverses, which corresponds to a periodic non-autonomous system, the blunted and the parallelepiped methods perform superior to the QR method, which in turn is superior to the naive interval and naive Taylor model methods.

5.3. **The Henon Map.** The discrete dynamics of the repeated application of the Henon map is a frequently used elementary example that exhibits many of the well-known effects of nonlinear dynamics, including chaos, periodic fixed points, islands and symplectic motion. The dynamics is two-dimensional, and given by

$$\begin{aligned} x_{n+1} &= 1 - \alpha x_n^2 + y_n \\ y_{n+1} &= \beta x_n. \end{aligned} \tag{5.3}$$

It can easily be seen that the motion is area preserving for $|\beta| = 1$.For our study, we borrow an example from the work of Kühn[19] illustrating the performance of the zonotope method and compare with TMs using shrink wrapping. We consider the dynamics for the special cases of $\alpha = 2.4$ and $\beta = -1$, and concentrate on initial boxes of the from $(x_0, y_0) \in (0.4, -0.4) + [-d, d]^2$. As an example to assess the dynamics, we consider the box with $d = 10^{-2}$ and study its evolution for a few turns.

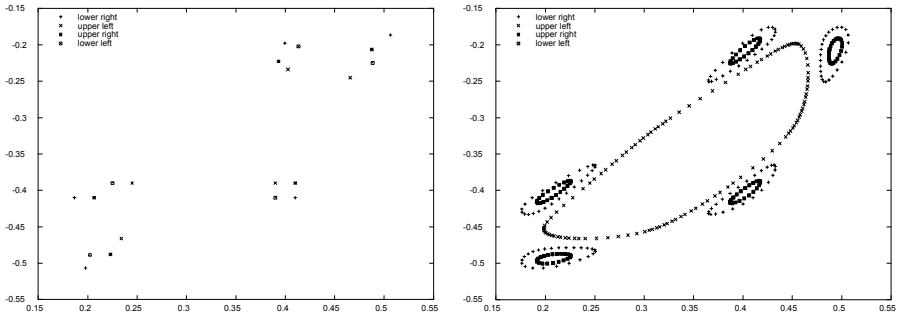

FIGURE 20. Iteration through the Henon map. Shown are the motion of the corner points of the box $(x_0, y_0) \in (0.4, -0.4) + [-d, d]^2$ for $d = 10^{-2}$ for five iterations (left) and for 120 iterations (right).

Figure 20 shows the motion of the four corner points for five iterations and for 120 iterations. It becomes apparent that three of the corner points are trapped in a five-fold island structure, while one of them follows an ergodic curve inside the

islands. This situation makes very long-term validated integration impossible since the transition region between the islands and the ergodic part is chaotic. As a first test, we study the dynamics of the box $(x_0, y_0) \in (0.4, -0.4) + [-d, d]^2$ for $d = 10^{-2}$ with first order Taylor models with shrink wrapping and compare with the results obtained by tenth order Taylor models with shrink wrapping; the results are shown in figure 21. It can be seen that the presence of the nonlinearities in the dynamics makes the size of the enclosures obtained by the linear method increase quickly. On the other hand, the higher order method can follow the details of the dynamics, including the "pulling apart" of the corner points rather well.

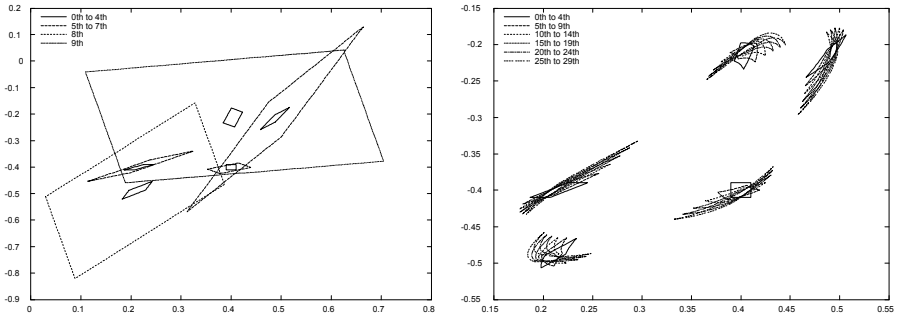

FIGURE 21. Dynamics through the Henon map for the box $(x_0, y_0) \in (0.4, -0.4) + [-d, d]$ for $d = 10^{-2}$ for nine turns with first order (left) and for 29 turns with tenth order (right) Taylor models with non-blunted shrink wrapping.

As a first example to study long term motion, we show the predicted inclusion after 500 iterations of the map for the case $d = 10^{-6}$. This choice of d entails that the entire box stays confined within the island structure, and is at least not subject to chaotic motion. Figure 22 shows the results obtained by the zonotope method, linear maps from $R^{m \cdot n}$ into R^n, for various numbers of of the parameter m and Taylor model methods of orders 1 and 5 using shrink wrapping. On the left, the results obtained by the Taylor model methods are overlaid on the respective results of the zonotope method; the picture was taken from [19]. For the purpose of better comparison, the TM results are also shown separately on the right. We see that the enclosure by the TM method is similarly accurate, and perhaps slightly sharper, than that of the zonotope method with $m = 15$. The right picture reveals that the TM method of order 5 produces a slightly sharper result than the TM method of order 1.

In passing we note that the values for the center point reported for the zonotope method in [19] are incorrect; in fact, the values provided there agree to all digits shown to those after 3 iterations, but not even to one digit with those 500 iterations, which because of the five-fold repetitive structure of the Henon map should be close to the starting point. However, because of the high degree of similarity of the $m = 15$ zonotope enclosure with that of the TM method after 500 iterations and the dissimilarity after 3 iterations, it appears very likely that the enclosure itself is indeed provided correctly. In order to study the behavior of the TM methods for long term problems, we iterate the map until failure occurs. In [19] it is reported

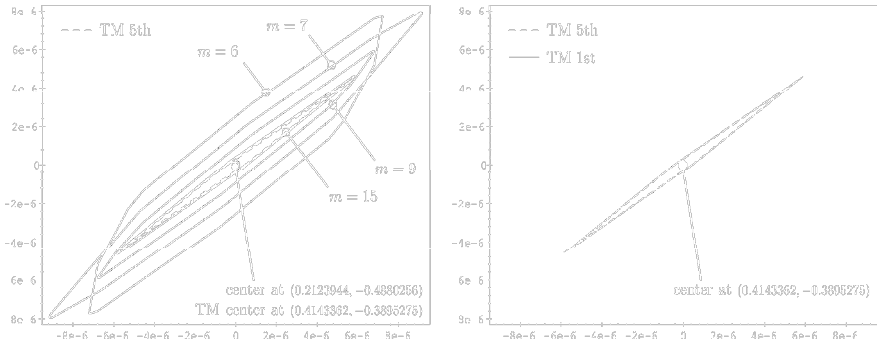

FIGURE 22. Validated enclosures after 500 iterations of the initial condition $(0.4, -0.4) + [-10^{-6}, 10^{-6}]^2$ through the Henon map. Shown are enclosures by the zonotope method of various values of the parameter m and by the TM methods of orders 1 and 5 using shrink wrapping (left). For better comparison, the results of the TM methods are also shown separately (right).

that for the domain $(x_0, y_0) \in (0.4,\ -0.4) + [-d, d]^2$ with $d = 10^{-12}$, the $m = 15$ zonotope method succeeds for about $33,000$ iterations. We compare this behavior with the preconditioned TM method of order 5 with shrink wrapping and observe that the method can succeed to provide enclosures for about $280,000$ iterations for order 5 and slightly longer for order 10. The TM method of order 1 survives for about $20,000$ iterations. Figure 23 shows some results of these computations. On the left we show the size of the remainder bounds for each turn, which is nonzero if the shrink wrapping fails to be executed. The remainder terms are usually in the range of 10^{-12}, but occasionally exceed 10^{-9}. The right shows the total accumulated shrink wrap factor, which is a measure of the inflation of the box. The seemingly large value of 10^6 is due to the fact that because of the proximity to the floating point floor, the initially small box size of 10^{-12} increases quickly. Approximately at the number of iterations at which the zonotope method fails to proceed, the shrink wrap factor stabilizes at about 10^6, leading to an overall box size of around 10^{-6}.

It is also interesting to study how much of an improvement shrink wrapping provides compared to iteration with naive Taylor models. Fig. 24 shows the remainder bounds obtained in this approach, and it is apparent that failure now occurs much more rapidly at around $16,000$ iterations, about half as much as the zonotope method is able to succeed.

In order to assess the expected influence of double precision floating point error, we attempt to simulate the behavior in quadruple precision. Due to the absence of an arbitrary precision or quadruple precision implementation of our TM tools, we perform a non-validated experiment in which the floating point accuracy threshold ε_m that is used in the internal interval operations was artificially set to the 10^{-30}, a number typical for the use of quadruple precision arithmetic. While the resulting inclusions are of course not validated results since the actual accuracy remains at the level of 10^{-15} or so, the results provide a rather good estimate for the growth of errors that is to be expected in quadruple precision.

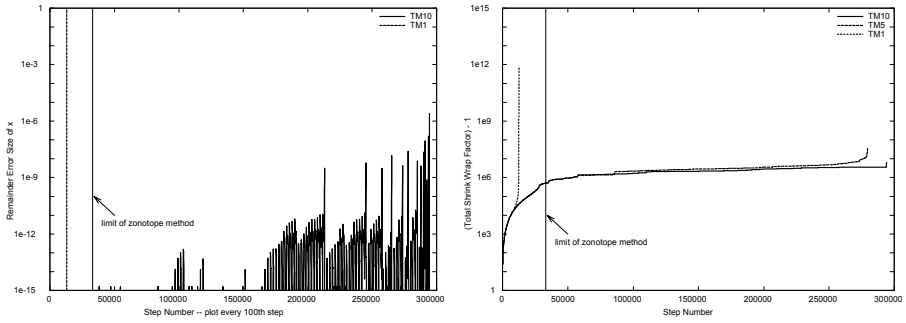

FIGURE 23. Dynamics in the Henon map for $(x_0, y_0) \in (0.4, -0.4) + [-d, d]^2$ with $d = 10^{-12}$. Shown are the remainder bounds (left) and shrink wrap factors (right) for TMs of order 1, 5, and 10.

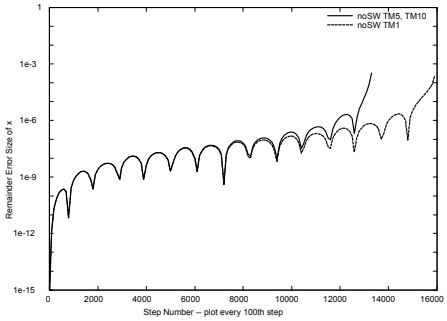

FIGURE 24. Dynamics in the Henon map for $(x_0, y_0) \in (0.4, -0.4) + [-d, d]^2$ with $d = 10^{-12}$ using naive Taylor models without shrink wrapping. Shown are the the remainder bounds for TMs of order 1, 5, and 10.

Repeating the study in this way, we observe that the survival time of the first order method now increases to a respectable $150,000$ iterations. But on the other hand, the higher order methods can now execute more than $7,500,000$ iterations, or about 50 times as much. A more detailed study of the results in figure 25 shows that beyond well over one million turns, the shrink wrap factor grows very moderately to about 10^{-6}, until just before 2 million turns, the first intermediate failures of shrink wrapping occur. At this point, the shrink wrap factor increases appreciably to re-absorb the remainder term a few iterations later, the map again becomes shrinkable. Overall it is clear that here the use of the higher order methods quite significantly improves performance, which seems to be limited mostly by floating point errors.

5.4. **A Muon Cooling Ring.** In this section we study a problem from the field

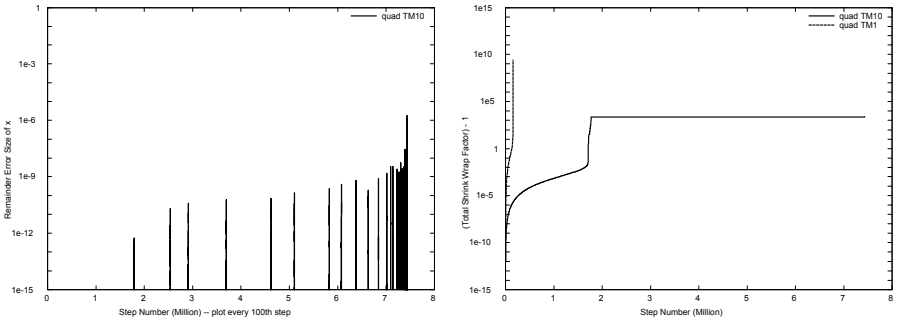

FIGURE 25. Non-validated dynamics in the Henon map for floating point errors similar to those in quadruple precision for $(x_0, y_0) \in (0.4, -0.4) + [-d, d]^2$ with $d = 10^{-12}$ Shown are the the remainder bounds (left) and shrink wrap factors (right) for TMs of order 1, 5, and 10.

of beam physics and illustrate the use of curvilinear coordinates. We use a simple model of a muon cooling ring, the purpose of which is to reduce the size of a beam by passing it through material and simultaneously accelerating it. Specifically, the particles are held in a confined orbit by a homogenous magnetic field in vertical direction; for reasons of simplicity, we restrict the dynamics to lie only in the plane. The coordinates describing the motion are the Euclidean x and y, and the corresponding momenta p_x and p_y.

The particles are moving in homogenous matter, which provides a deceleration force of magnitude α along their direction of motion; the direction of motion is given by $(p_x, p_y)/\sqrt{p_x^2 + p_y^2}$. Furthermore, there is an azimuthal acceleration force of equal magnitude α and opposite direction. For particles at coordinates (x, y), the azimuthal direction is given by $(y, -x)/\sqrt{x^2 + y^2}$. Altogether, the equations of motion are

$$\begin{aligned} \dot{x} &= p_x \\ \dot{y} &= p_y \\ \dot{p}_x &= p_y - \frac{\alpha}{\sqrt{p_x^2 + p_y^2}} \cdot p_x + \frac{\alpha}{\sqrt{x^2 + y^2}} \cdot y \\ \dot{p}_y &= -p_x - \frac{\alpha}{\sqrt{p_x^2 + p_y^2}} \cdot p_y - \frac{\alpha}{\sqrt{x^2 + y^2}} \cdot x \end{aligned} \tag{5.4}$$

It can be easily verified that the system has an invariant solution

$$(x, y, p_x, p_y)_I(t) = (\cos t, -\sin t, -\sin t, -\cos t),$$

which represents a clockwise rotation in the horizontal plane with constant radius 1 and constant momentum 1. The practically significant property of the system is that acceleration always happens azimuthally, while deceleration happens in the

direction of motion; this leads to a decrease of the radial component of the momentum, and mathematically to the fact that solutions of the ODE asymptotically approach circular motion of the form

$$(x, y, p_x, p_y)_a(t) = (\cos(t-\phi), -\sin(t-\phi), -\sin(t-\phi), -\cos(t-\phi)),$$

where ϕ is a characteristic angle of the particle in question. For practical applications, this is eminently useful, as it reduces the volume of the four dimensional space of values (x, y, p_x, p_y), an effect known as cooling. While in practice, many technical details have to be considered, the simple ODE (5.4) represents the essence of this process.

For the purpose of using the problem as a test case for validated integration, the following aspects are important

(1) It is important to treat a large initial domain box of a range of $[-10^{-2}, 10^{-2}]^4$. This will entail the presence of rather strong nonlinearities.
(2) Because of the transversal damping action towards the invariant limit cycle, the linear part of the motion will be more and more ill-conditioned.

We study the dynamics using COSY-VI using curvilinear preconditioning, which is standard in beam physics simulations (see for example [26], [3] and references therein). We perform the integration until no further reduction in phase space can be performed due to the proximity of the floating point floor. Figure 26 shows the effects of cooling for domain boxes $(0, 1, 1, 0) + [-d, d]^4$ for $d = 10^{-2}$, 10^{-4}, and 10^{-6}; the value of 10^{-2} approximately corresponds to the practical needs.

Studying the magnitude of the determinants of the linear part, which roughly correspond to the volume, we see that cooling happens exponentially and with nearly the same speed in all three cases. The final volume that is attained is larger for the larger initial volumes, which is due to the fact that volume gets compressed only transversely to the motion of the beam, while nothing affects the particle's longitudinally motion. Thus for larger initial boxes, the final box in the direction of motion will be larger.

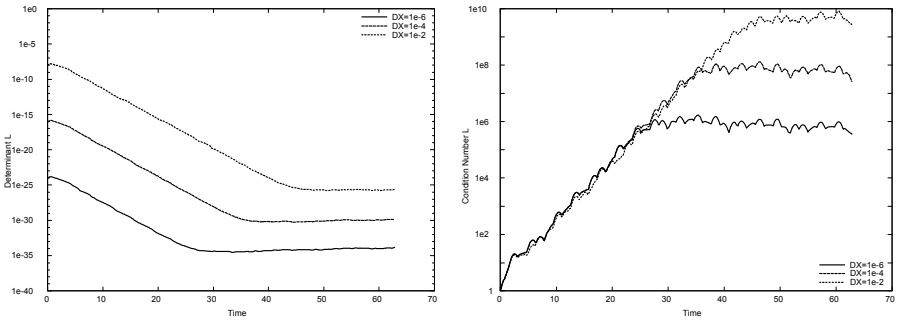

FIGURE 26. Simulation of the Muon Cooling Ring for initial condition boxes $(0, 1, 1, 0) + [-d, d]^4$ located at the upper top for $d = 10^{-2}$, 10^{-4}, 10^{-6}. Shown are the determinants of the linear part (left) indicating progress in cooling, and the condition numbers of the linear part (right).

As a result, we obtain a very narrow elongated structure with nearly vanishing radial thickness that rotates around a circle. As a consequence, the condition number of the linear part becomes larger and larger, as shown on the right of figure 26. If not treated properly, computationally this may represent a significant challenge, but as expected, the curvilinear preconditioning can overcome this difficulty. To study the motion in detail, we look at the remainder bounds of the dynamics, which are shown in 27. Overall, we see that COSY-VI has no difficulty performing the

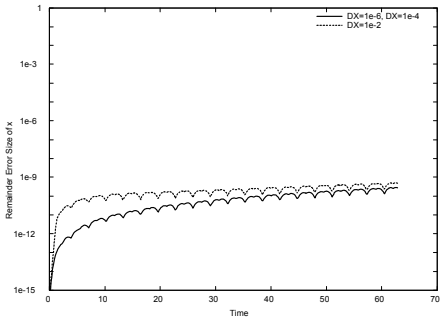

FIGURE 27. Remainder bound sizes for the simulation of the Muon Cooling Ring for initial condition boxes $(0,1,1,0)+\ [-d,d]^4$ for $d = 10^{-2}$, 10^{-4}, 10^{-6}.

integration of the muon damping system with $d = 10^{-2}$ for ten revolutions, which is sufficient to perform the required damping task. On the other hand, the linear code AWA can only succeed with this task for $d = 10^{-4}$. Thus a full simulation of the necessary space of initial conditions, which can be performed with one run of COSY-VI, would require approximately $\left(10^2\right)^4 = 10^8$ runs of AWA.

5.5. **The Discrete 2D Circular Kepler Problem.** This system describes the dynamics of circular Kepler orbits around a central mass in terms of the variables (x, y) in the plane of the motion. It is well known from Kepler's third law that the periods T and large semi-major axes a of a Kepler ellipse are related via $T^2 = k \cdot a^3$, where k is determined by the mass of the central object. For circular orbits of radius r, for $k = 1$ this entails an angular velocity of $\omega = 2\pi/T = 2\pi \cdot r^{-3/2}$, which means that the transformation by a fixed time step Δt is given by the two-dimensional transformation

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} \cos\Delta\phi & \sin\Delta\phi \\ -\sin\Delta\phi & \cos\Delta\phi \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix}$$
$$\text{where } \Delta\phi = \frac{2\pi\Delta t}{\left(x^2+y^2\right)^{3/4}}.$$

While addressing only circular motion, the dynamics is also quite characteristic of the general motion of Kepler ellipses because it captures one of the main effects: as time progresses, there is a larger and larger lag between the circles of different radii r. This lag makes Taylor expansion of final condition in terms of initial conditions

impossible for sufficiently large times, and thus represents a challenge for all Taylor-based methods that will necessarily lead to their eventual failure. The interest in the problem now lies in the attempt to delay failure.

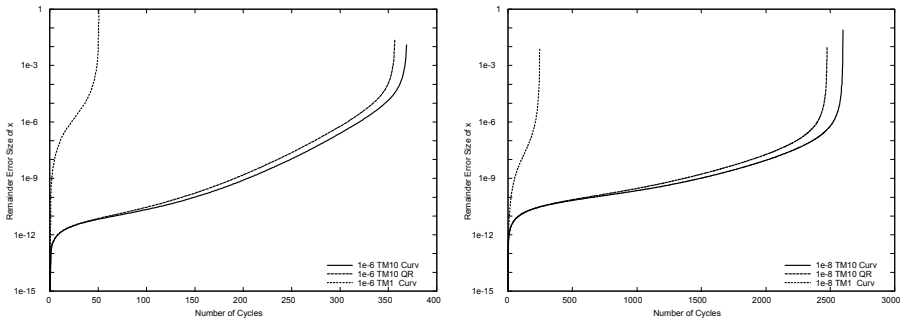

FIGURE 28. Dynamics in the discrete 2D Kepler system for initial box sizes widths of 10^{-6} (left) and 10^{-8} (right). Shown are the remainders obtained by the first and tenth order Taylor method using Curvilinear Preconditioning and QR preconditioning.

Figures 28 and 28 show the remainder bounds of the study of the dynamics without shrink wrapping for repeated application of the discrete transformation with $\phi_0 = \pi/4$, in which case one full revolution, or one cycle, consists of eight applications of the individual map.

We study three cases: as a reference we use first order Taylor models preconditioned by curvilinear coordinates, which behave similar to the PEQR method. We compare with tenth order Taylor models preconditioned by curvilinear coordinates, and tenth order Taylor models preconditioned by the QR method. The growth of the remainder bounds is shown for four different initial domain widths of 10^{-6}, 10^{-8}, 10^{-10}, and 10^{-12} as a function of full cycles of 2π. It can be seen that for each case, the tenth order Taylor model method survives between 7 and 10 times longer than the first order method. Furthermore, the preconditioning by curvilinear coordinates leads to a slightly better performance, which is attributed to the fact that the movement of the coordinate system is smoother since it follows the reference orbit instead of the somewhat more random orientation of the longest edge.

It is also interesting to estimate the growth rate of the remainder bounds in the high-order TM methods. An inspection of the right picture in figure 29 reveals that during revolutions 1000 and 6000, the remainder width increases from about 10^{-10} to about 10^{-9}, for a total increase of $9 \cdot 10^{-9}$ over $5,000$ revolutions or $40,000$ iterations. This corresponds to about $2 \cdot 10^{-13}$ per map iteration; considering that each iteration requires several function evaluations, and that in our current implementation, intrinsic functions carry an overestimation of around 10 ulps, this

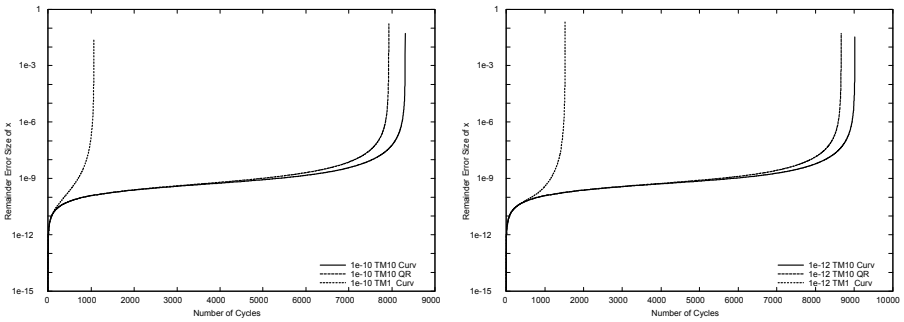

FIGURE 29. Dynamics in the discrete 2D Kepler system for initial box sizes widths of 10^{-10} (left) and 10^{-12} (right). Shown are the remainders obtained by the first and tenth order Taylor method using Curvilinear Preconditioning and QR preconditioning.

number is very close to the unavoidable consequences of accounting for the mere floating point errors of the arithmetic involving the constant part of the Taylor model.

5.6. **Acknowledgement.** We thank many colleagues for various stimulating discussions, in particular Ramon E. Moore, and also Rudolf Lohner, Ken Jackson, Ned Nedialkov, Markus Neher, George Corliss, and John Pryce. The work was supported in part by the Illinois Consortium for Accelerator Research and in part by the US Department of Energy, Grant Number DE-FG02-95ER4093, the National Science Foundation, and an Alfred P. Sloan Fellowship.

REFERENCES

[1] W. F. Ames and E. Adams. Monotonically convergent numerical two-sided bounds for a differential birth and death process. In K. Nickel, editor, *Interval Mathematics*, volume 29 of *Lecture Notes in Computer Science*, pages 135–140, Berlin; New York, 1975. Springer-Verlag.
[2] C. Barbarosie. Reducing the wrapping effect. *Computing*, 54:347–357, 1995.
[3] M. Berz. *Modern Map Methods in Particle Beam Physics*. Academic Press, San Diego, 1999. Also available at http://bt.pa.msu.edu/pub.
[4] M. Berz, J. Hoefkens, and K. Makino. COSY INFINITY Version 8.1 - programming manual. Technical Report MSUHEP-20703, Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48824, 2002. see also http://cosy.pa.msu.edu.
[5] M. Berz and K. Makino. Verified integration of ODEs and flows using differential algebraic methods on high-order Taylor models. *Reliable Computing*, 4(4):361–369, 1998.
[6] M. Berz and K. Makino. New methods for high-dimensional verified quadrature. *Reliable Computing*, 5(1):13–22, 1999.
[7] M. Berz and K. Makino. Preservation of canonical structure in nonplanar curvilinear coordinates. *International Journal of Applied Mathematics*, 3,4:401–419, 2000.
[8] G. F. Corliss. Survey of interval algorithms for ordinary differential equations. *Appl. Math. Comput.*, 31:112–120, 1989.
[9] G. F. Corliss. Where is validated ODE solving going? 1996.
[10] P. Eijgenraam. The solution of initial value problems using interval arithmetic. *Mathematical Centre Tracts*, No. 144, 1981.

[11] B. Erdélyi and M. Berz. Optimal symplectic approximation of Hamiltonian flows. *Physical Review Letters*, 87,11:114302, 2001.
[12] B. Erdélyi and M. Berz. Local theory and applications of extended generating functions. *International Journal of Pure and Applied Mathematics*, 11,3:241–282, 2004. available at http://bt.pa.msu.edu/pub.
[13] T. Gambill and R. Skeel. Logarithmic reduction of the wrapping effect with application to ordinary differential equations. *SIAM J. Numer. Anal.*, 25:153–162, 1988.
[14] J. Hoefkens, M. Berz, and K. Makino. Controlling the wrapping effect in the solution of ODEs of asteroids. *Reliable Computing*, 9(1):21–41, 2003.
[15] L. W. Jackson. A comparison of ellisoidal and interval arithmetic error bounds. *SIAM Review*, 11:114, 1969.
[16] L. W. Jackson. Dept. of Computer Science, University of Toronto, 1971.
[17] L. W. Jackson. Interval arithmetic error bounding alorithms. *SIAM Journal of Numerical Analysis*, 12:223, 1975.
[18] W. M. Kahan. Circumsribing an ellipsoid about the intersection of two ellipsoids. *Candian Mathematical Bulletin*, 11:437, 1968.
[19] W. Kühn. Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61:47–67, 1998.
[20] R. J. Lohner. AWA - Software for the computation of guaranteed bounds for solutions of ordinary initial value problems.
[21] R. J. Lohner. Enclosing the solutions of ordinary initial and boundary value problems. In E. Kaucher, U. Kulisch, and C. Ullrich, editors, *Computer Arithmetic: Scientific Computation and Programming Languages*, pages 255–286. Teubner, Stuttgart, 1987.
[22] R. J. Lohner. *Einschliessung der Losung gewohnlicher Anfangs- und Randwertaufgaben und Anwendungen.* Dissertation, Fakultat fur Mathematik, Universitat Karlsruhe, 1988.
[23] R. J. Lohner. *Einschliessungen bei Anfangs- und Randwertaufgaben gewohnlicher Differentialgleichungen; and: Praktikum 'Einschliessungen bei Differentialgleichungen',*, pages 183–207 and 209–223. Akademie-Verlag, Berlin, 1989.
[24] R. J. Lohner. Computation of guaranteed enclosures for the solutions of ordinary initial and boundary value problems. In J. Cash and I. Gladwell, editors, *Computational Ordinary Differential Equations*, pages 425–435. Clarendon Press, Oxford, 1992.
[25] R. J. Lohner. Step size and order control in the verifed solution of ivp with ode's. In *SciCADE '95 International Conference on Scientific Computation and Differential Equations*, pages 28.3.–1.4., Stanford, 1995.
[26] K. Makino. *Rigorous Analysis of Nonlinear Motion in Particle Accelerators.* PhD thesis, Michigan State University, East Lansing, Michigan, USA, 1998. Also MSUCL-1093.
[27] K. Makino and M. Berz. Remainder differential algebras and their applications. In M. Berz, C. Bischof, G. Corliss, and A. Griewank, editors, *Computational Differentiation: Techniques, Applications, and Tools*, pages 63–74, Philadelphia, 1996. SIAM.
[28] K. Makino and M. Berz. Efficient control of the dependency problem based on Taylor model methods. *Reliable Computing*, 5(1):3–12, 1999.
[29] K. Makino and M. Berz. Perturbative equations of motion and differential operators in non-planar curvilinear coordinates. *International Journal of Applied Mathematics*, 3,4:421–440, 2000.
[30] K. Makino and M. Berz. The method of shrink wrapping for the validated solution of odes. Technical Report MSUHEP-20510, Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48824, 2002.
[31] K. Makino and M. Berz. New applications of Taylor model methods. In G. Corliss, C. Faure, A. Griewank, L. Hascoët, and U. Naumann, editors, *Automatic Differentiation of Algorithms from Simulation to Optimization*, pages 359–364. Springer, 2002.
[32] K. Makino and M. Berz. Taylor models and other validated functional inclusion methods. *International Journal of Pure and Applied Mathematics*, 6,3:239–316, 2003. available at http://bt.pa.msu.edu/pub.
[33] R. E. Moore. Automatic local coordinate transformation to reduce the growth of error bounds in interval computation of solutions of ordinary differentical equations. In L. B. Rall, editor, *Error in Digital Computation, Vol II*, 1965.
[34] R. E. Moore. *Interval Analysis.* Prentice-Hall, Englewood Cliffs, NJ, 1966.
[35] R. E. Moore. *Methods and Applications of Interval Analysis.* SIAM, 1979.

[36] N. S. Nedialkov. *Computing Rigorous Bounds on the Solution of an Initial Value Problem for an Ordinary Differential Equation*. PhD thesis, University of Toronto, Toronto, Canada, 1999.

[37] N. S. Nedialkov and K. R. Jackson. Methods for initial value problems for ordinary differential equations. In U. K. et al., editor, *Perspectives on Enclosure Methods*, pages 219–264. Springer, Berlin, 2001.

[38] N. S. Nedialkov and K. R. Jackson. A new perspective on the wrapping effect in interval methods for IVPs for ODEs. *Proc. SCAN2000, Kluwer*, 2001.

[39] N. S. Nedialkov, K. R. Jackson, and G. F. Corliss. Validated solutions of initial value problems for ordinarly differential equations. *Appl. Math. Comput.*, 105:21–68, 1999.

[40] A. Neumaier. The wrapping effect, ellipsoid arithmetic, stability and confidence regions. *Computing Supplementum*, 9:175–190, 1993.

[41] N. Revol, K. Makino, and M. Berz. Taylor models and floating-point arithmetic: Proof that arithmetic operations are validated in COSY. *Journal of Logic and Algebraic Programming*, in print, 2004. University of Lyon LIP Report RR 2003-11, MSU Department of Physics Report MSUHEP-30212, http://bt.pa.msu.edu/pub.

[42] N. F. Stewart. A heuristic to reduce the wrapping effect in the numerical solution of ODEs. *BIT*, 11:328–337, 1971.

6.

Department of Physics, University of Illinois at Urbana-Champaign, 1110 W. Green Street, Urbana IL 61801-3080, USA

Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48824, USA