.

3. Taylor Model Integrators

Next we discuss how the Taylor model based integrator models $R(t)$ for the next time step. The algorithm describes $R(t)$ by a Taylor model vector that depends on the initial condition $\vec{x}_0$. During computation in a time step, the Taylor model arithmetic includes the dependency on time t, as indicated in the Picard operation in (2.3). At the completion of the time step, a numerical value of time t is inserted, so only the dependency on the initial condition $\vec{x}_0$ remains. In the case of linear ODEs, the structure of the parallelepiped $R(t)$ is kept as it is as shown in (f) in Figure 1, and in practice, the Taylor model vector contains a very small remainder interval vector arisen in the process of the Picard operation in (2.3). In the case of nonlinear ODEs, the shape of $R(t)$ is optimally preserved as shown in (f) in Figure 2, namely approximated by a high order Taylor polynomial vector $\vec{P}$ depending on $\vec{x}_0$, where the accuracy is easily raised by the computational order of Taylor models. There is no special trick necessary. We first set up the initial condition for the ODEs in the form of Taylor models as (2.2). Then, via the Taylor model arithmetic in the Picard operation in (2.3) in the Schauder's fixed point iteration algorithm [**?**], we automatically obtain a Taylor model solution at time t, which depends on the initial condition $\vec{x}_0$;

$$\forall \vec{x}_0 \in \vec{B}, \quad \vec{x}(\vec{x}_0, t) \in \vec{P}(\vec{x}_0) + \vec{I}. \tag{3.1}$$

The true solution is tightly enclosed in the Taylor model vector; the bulk behavior of the solution is represented by the high order Taylor polynomial vector $\vec{P}$, and the solution is verified by the remainder bound interval vector $\vec{I}$. The next time step computation takes the resulting Taylor model vector (3.1) as the starting value, and altogether there is no need for re-packing a solution.

4. Faithful Representation of the Flow - Taylor Model Wrapping

The flow representation is the most significant in the sense that it cannot be affected by a reduction of step, increase of order or any other method for increasing the accuracy of the local step; it can also not be affected by increases of the accuracy of floating number representation. It is directly connected to and characteristic of the mathematical behavior of the ODE being studied. We observe that for linear systems, the 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 dynamical values 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.

5. A Numerical Example - The Volterra Equations

We discussed and illustrated various methods to cope with the wrapping effect. In this section, we illustrate the difference between the conventional methods and the Taylor model method using a numerical example of a simple nonlinear ODE system. We use AWA [**?**] to represent the conventional methods, since it is one of

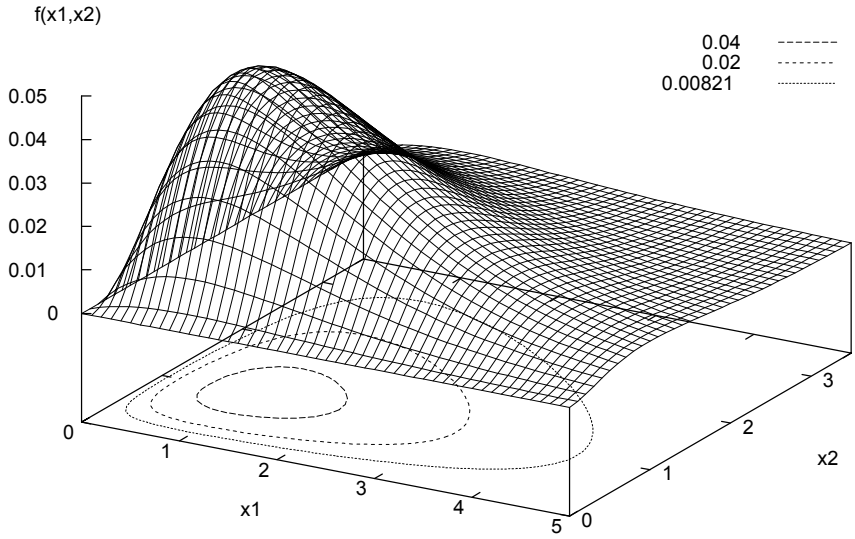

FIGURE 3. The solution trajectories of the Volterra equations, $f(x_1, x_2) \equiv x_1 x_2^2 e^{-x_1-2x_2} = c$(onstant). $f(x_1, x_2)$ is shown by mesh, and the contour lines for various c's are shown. The initial condition $(x_{01}, x_{02}) = (1, 3)$ corresponds to $c = 9e^{-7} \simeq 0.00821$.

the most successful codes and is widely spread. Our Taylor model based integrator COSY-VI is coded in COSY Infinity [**?**].

5.1. The Volterra Equations. The ODEs are the Volterra equations governing the growth of two conflicting populations, modeling a prey-predator relation. By the nature of the underlying problem, the values of initial condition as well as the solutions are positive, and the solution trajectories form closed orbits. The period of one cycle depends on the initial condition, and outer orbits take longer. However, if any component of the initial condition is negative, the solution trajectory diverges. It is a frequently cited example for the numerical verification of ODE solvers. If the method works, the solution enclosure develops like a closed band for one cycle, provided all the components of the initial condition are positive. Hence the problem serves as a good test case for validated ODE solvers, including the study of wrapping effect.

We take the same model discussed by Ames and Adams [**?**] and by Moore [**?**], 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 the conventional approaches. The ODEs and initial conditions for the Volterra equations are

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

The right hand side of the ODEs has the form of a "single use expression" (SUI), so it has no source of overestimation of arithmetic nature; this makes any overestimation due to the wrapping effect more clearly visible.

The solution trajectory follows the constraint

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

as can be seen by simple differentiation and insertion of the ODE, forming closed orbits if both x_{01} and x_{02} are positive, as shown by the contour lines in Figure 3. The solution trajectory for the point initial values $(x_{01}, x_{02}) = (1, 3)$ is a closed orbit with the period about $T \simeq 5.488138468035$. We carried out the integration of the system by AWA and COSY-VI, demanding one cycle period T. As will be shown, the system starts to show noticeable nonlinearity around $t \sim 4$.

5.2. AWA and COSY-VI. We used AWA with its standard mode; namely we use the enclosure method 4 that is the 'intersection of interval-vector and QR-decomposition' [**?, ?, ?**]. If the ODEs are linear, the enclosure method 4 performs the integration like the picture (f) in Figure 1. Thus, even for nonlinear ODEs, if the dependence of the solution to the initial condition is almost linear, the method often performs rather well. 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 [**?**], as we will see later.

The computational order has to be pre-set in both AWA and COSY-VI, and the same order setting was used for the direct comparison, even though the meaning of the computational order differs due to the different concepts and algorithms. In the case of AWA, it is the order of the Taylor expansion in time t. In the case of COSY-VI, it is the order of the Taylor models (1.1) throughout the whole computation including the process of the Picard operation in (2.3), thus the Taylor polynomial is expanded in time and the initial conditions up to the specified order.

Both AWA and COSY-VI have automatic step size control. AWA requires only the initial starting step size, and we chose the option to let AWA decide it automatically. Once the error exceeds the control threshold, AWA typically reduces the step size without limit, resulting in breaking down soon, as observed in the picture of step size progress in Figure 4. COSY-VI has maximum and minimum step size limits addition to the suggested starting step size.

In this example of the Volterra equations, our aim is to demonstrate how the Taylor model approach can represent the flow faithfully while suppressing the wrapping effect automatically. Because of this, we do not use shrink wrapping here.

At the completion of the integration period $T = 5.488138468035$, the global error was as low as 0.0006 and 0.005 for the 18th and the 12th order COSY-VI computation. We note, despite of the slightly different mechanism of the step size control, the behavior of step size progress has some similarities between AWA and COSY-VI, as is to be expected.

5.3. Solution Enclosures through the Integration Process. We performed the integration of the Volterra equations by AWA and COSY-VI with various computational orders, demanding the completion of one cycle period of the center point, $T = 5.488138468035$.

The pictures in Figure 5 show the solution regions $R(t)$ at various characteristic times, as they are enclosed by Taylor models. Initially nonlinearity is not very

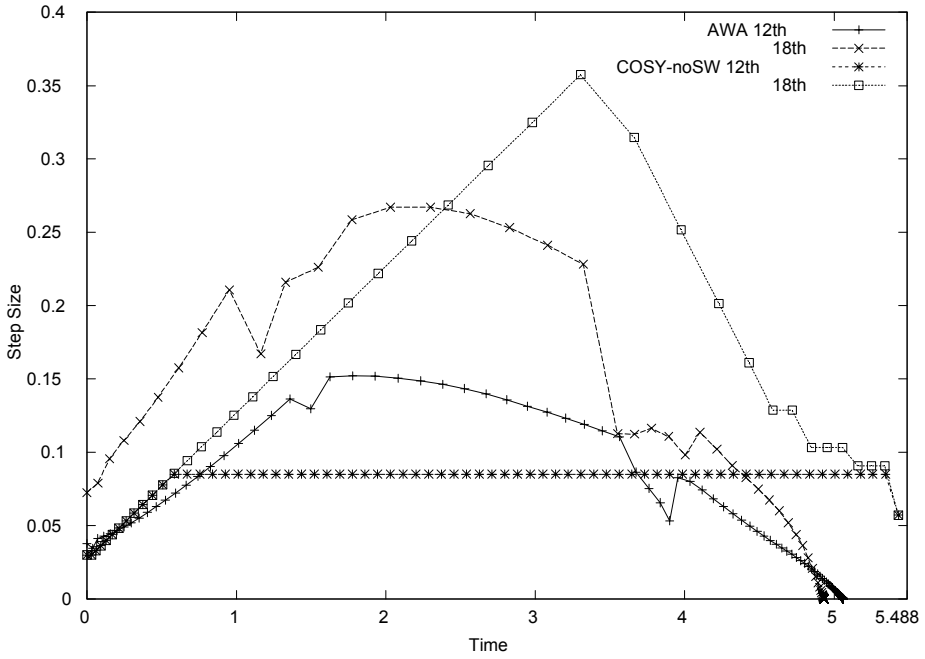

FIGURE 4. The progress of step size by AWA and COSY-VI (without shrink wrapping).

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. The nonlinearity temporarily decreases afterward, but the strong nonlinearity comes back just before the completion of the period, shown by $R(5.45)$.

The solution enclosures at each time step through the 18th order Taylor model computation by COSY-VI are placed along the center point trajectory in Figure 6. 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, and that is the result of different cycle periods for the various closed orbits. The dashed boxes are 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 6 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 5.

On the other hand, COSY-VI continues the computation during the period of strong nonlinearity by keeping the step size smaller, and when the nonlinearity becomes weak again, the step size increases again, provided the global error is

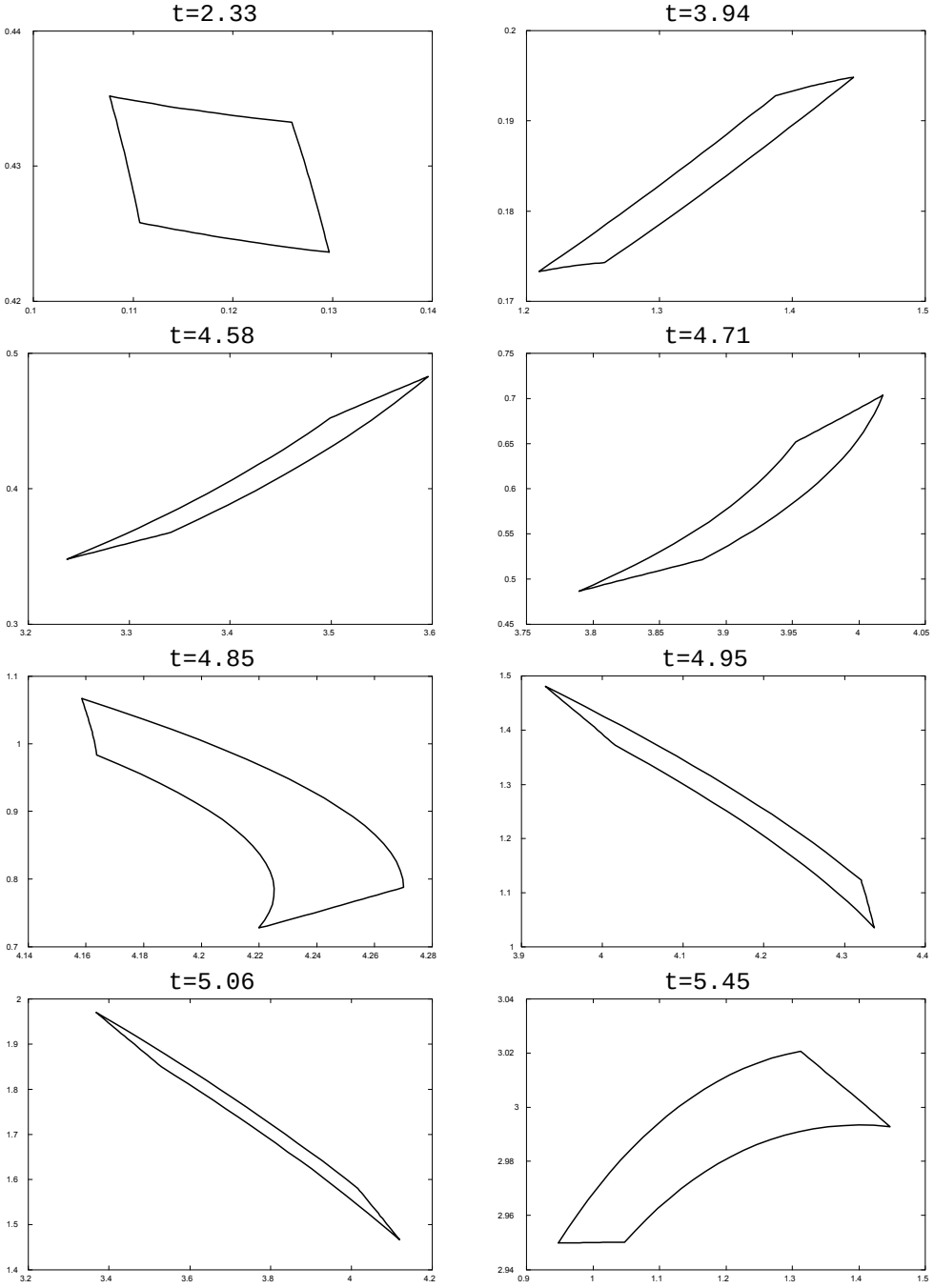

FIGURE 5. Solution enclosures at characteristic times, obtained by COSY-VI with computation order 18.

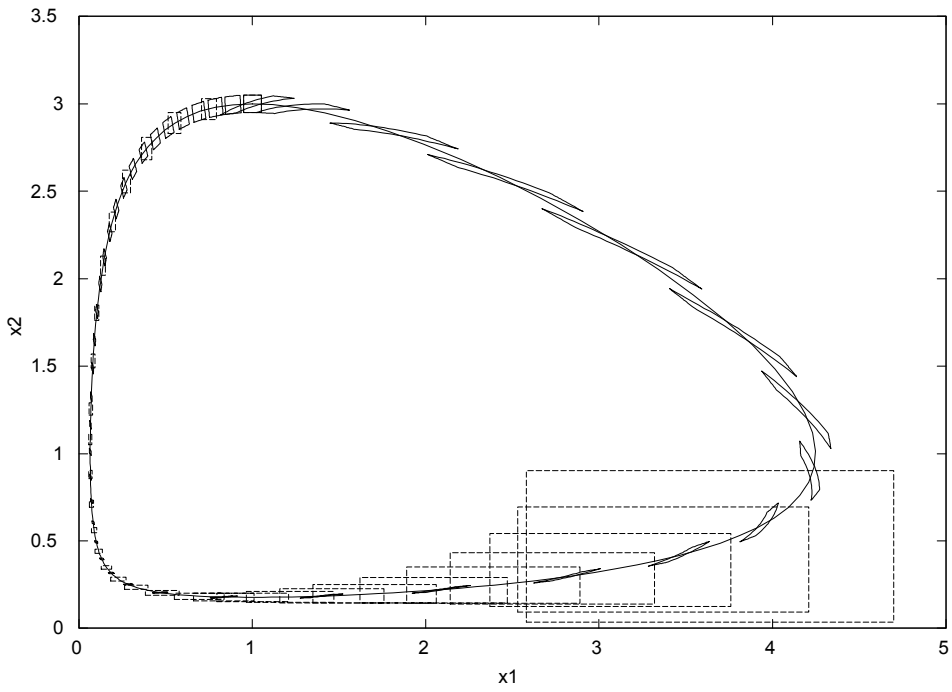

FIGURE 6. Solution enclosures of the Volterra eqs. at each time step by Taylor models (solid regions) and AWA (dashed boxes) with the 18th order computation.

still much lower than the control function. When the step size control is done only connected to the the local error, the step size progress directly indicates the difficulty of integration, hence the strength of nonlinearity. By applying shrink wrapping at every time step, the local error can be separated from the global error easily, and then the recovery of step size is often observed.

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 without shrink wrapping completed the whole demanded integration period T without difficulty when the computational order was high. For lower order computation, it was necessary to keep the step size small, as mentioned earlier. Also listed in Table 1 is the CPU time comparison, using a Pentium III Linux PC with 450 MHz. Since AWA did not complete the period, we also listed the breakdown time t in the ODE system. As it can be observed in the table and the step size progress in Figure 4, it is illuminating that the successful integration time t by AWA cannot be increased by simply increasing the computation order, and interestingly it is caused by the too fast growth of the step size in the beginning period for the higher order computation.

Order	COSY-VI	AWA	
	CPU time	CPU time	Breakdown time t
12	6.0 sec	13.6 sec	5.06039
18	25.5 sec	10.7 sec	4.93115

TABLE 1. CPU time. COSY-VI completed the whole integration period $T = 5.488138468035$, but AWA broke down at time t.

5.4. Local Time Step Errors and Rounding Errors. The error in the local time step can usually be decreased by decreasing the step size and/or increasing the order of the method; and apparently the situation is the same for the Taylor model integrator. These improvements continue until near the floor of the machine accuracy is reached. In most situations this has to be assumed as fixed, but of course one could also execute the same calculation with larger word length, or even utilize arbitrary precision arithmetic. For the purposes of our study, we ignore these two options at this point. In the case of the Taylor model integrator, at the beginning of the kth time step, we are given a Taylor model representing the flow of the ODE as

$$\vec{x}_k \in \vec{I} + \bigcup_{\vec{x}_0 \in \vec{B}} \vec{P}_k(\vec{x}_0)$$

where $\vec{P}_k$ is a polynomial in the initial coordinates $\vec{x}_0$ describing the transformation of the original box $\vec{B}$, and $\vec{I}$ is a remainder bound interval vector. This Taylor model enters the Picard operator of the ODE, which has the form

$$\vec{x}(t) = \mathcal{A}\left(\vec{x}(t)\right) = \vec{P}_k(\vec{x}_0) + \vec{I} + \int_{t_k}^{t_k+\Delta t} \vec{F}(\vec{x}\left(t'\right), t')\ dt'.$$

The next step consists in finding a Taylor model $\vec{P}_{k+1}(\vec{x}_0, t) + \vec{I}$ in the initial conditions $\vec{x}_0$ and time t and with domain of definition $\vec{B} \times [t_k, t_k + \Delta t]$ providing a self-inclusion of the operator $\mathcal{A}$, i.e. satisfying $\mathcal{A}(\vec{P}_{k+1}(\vec{x}_0, t) + \vec{I}) \subset \vec{P}_{k+1}(\vec{x}_0, t) + \vec{I}$ over all of $\vec{B} \times [t_k, t_k + \Delta t]$. As shown in detail in [**?**], this is sufficient to assert a solution of the ODE in $\vec{P}_{k+1}(\vec{x}_0, t) + \vec{I}$ for any point lying in the ; and in particular, the solution

At this point it is also worth mentioning that one may consider to perform new time steps by "re-starting" the flow with the identity map $\mathcal{I}$ instead of $\vec{P}_k(\vec{x}_0) + \vec{I}$, and once a solution is found, composing the solution for the $(k + 1)$st time step with the solution of the previous time steps to obtain the overall result. It is to be expected that the enclosure operation will be performed more easily in this approach since the Taylor model of $\vec{P}_k(\vec{x}_0) + \vec{I}$ cannot add computational complexity to the inclusion. However, except in cases with sufficient amount of damping, the method will not provide a guaranteed solution of the enclosure, since this would require that the previous time step result is contained in $\vec{B}$, which is in general not the case. To remedy the situation, it would be necessary to increase $\vec{B}$ such that it provides an enclosure for $\vec{P}(\vec{x}_0) + \vec{I}$, which unless $\vec{P}$ is near the identity, provides for a large amount of overestimation and, after repeated steps, a rapid growth of computational error. Thus this method is not pursued in the COSY-VI integration scheme. The attempt to provide an analysis of the error growth of the