Abstract: Taylor model methods represent a combination of high-order multivariate automatic differentiation and the simultaneous computation of an interval remainder bound enclosing approximation error over a given domain. This method allows a far-reaching suppression of the dependency problem common to interval methods, and can thus often be used for precise range bounding problems. We compare the performance of the method with other state of the art validated tools including centered forms and mean value forms. We also compare with the computation of remainder bounds via high-order interval automatic differentiation.

Key-Words: Multivariate, range bounding, interval, Taylor model, COSY Infinity.

1 Introduction

The Taylor model approach[1] has been developed as an augmentation to earlier work on high-order multivariate automatic differentiation and the differential algebraic methods to solve ODEs. Specifically, final variables in a code list are expressed in terms of a high-order multivariate floating point Taylor polynomial of initial variables, plus a remainder bound accounting for the approximation error. Over suitably small domains, the polynomial representation is naturally free of most of the dependency problem that the underlying function may have had. At each node of the code list, the remainder bound is calculated in parallel to the floating point coefficients; since this only requires information about the current Taylor coefficients, its calculation itself is also free of much of the dependency problem of the original code list.

The Taylor model methods capitalize on this obvious observation by representing any functional dependency in terms of a (Taylor) polynomial of sufficiently high order, plus a small interval bound capturing the parts of the function that deviate from the polynomial. As such it is merely a validated extension of automatic differentiation methods[2], namely those of high order in many variables [3].

In this paper, first we review an arithmetic of Taylor models and some algorithms that allow to perform a variety of common analytical operations, including efficient range bounding for global optimization. We will compare the behavior of Taylor models (TM) with those a variety of other tools and approaches for some of the typical applications. We will study the interval method (I), as well as the more advanced inclusion methods of the centered form (CF) and the mean value form (MF). We also compare with the method of using interval automatic differentiation (IAD) to compute a Taylor polynomial and a remainder bound.

To summarize previous work on Taylor models, the method has the following fundamental properties:

1. The ability to provide enclosures of any function given by a finite computer code list by a Taylor polynomial and a remainder bound with a sharpness that scales with order \( (n + 1) \) of the width of the domain

2. The ability to alleviate the dependency problem in the calculation
3. The ability to scale favorably to higher dimensional problems

More specifically, the following definitions and theorems outline the arithmetic and basic properties of the method.

**Definition 1 (Taylor Model)** Let \( f : D \subset \mathbb{R}^n \rightarrow \mathbb{R} \) be a function that is \((n+1)\) times continuously partially differentiable on an open set containing the domain \( D \). Let \( x_0 \) be a point in \( D \) and \( P \) the \( n \)th order Taylor polynomial of \( f \) around \( x_0 \). Let \( I \) be an interval such that

\[
 f(x) \in P(x - x_0) + I \text{ for all } x \in D.
\]

Then we call the pair \( (P, I) \) an \( n \)th order Taylor model of \( f \) around \( x_0 \) on \( D \).

Apparently \( P + I \) encloses \( f \) between two hypersurfaces on \( D \). As a first step, we develop methods to calculate Taylor models from those of smaller pieces.

**Definition 2 (Addition and Multiplication of Taylor Models)** Let \( T_{1,2} = (P_{1,2}, I_{1,2}) \) be \( n \)th order Taylor models around \( x_0 \) over the domain \( D \). We define

\[
 T_1 + T_2 = (P_1 + P_2, I_1 + I_2) \quad \text{and} \quad T_1 \cdot T_2 = (P_{1,2}, I_{1,2})
\]

where \( P_{1,2} \) is the part of the polynomial \( P_1 \cdot P_2 \) up to order \( n \) and

\[
 I_{1,2} = B(P_c) + B(P_1) \cdot I_2 + B(P_2) \cdot I_1 + I_1 \cdot I_2
\]

where \( P_c \) is the part of the polynomial \( P_1 \cdot P_2 \) of orders \((n + 1)\) to \( 2n \), and \( B(P) \) denotes a bound of \( P \) on the domain \( D \). We demand that \( B(P) \) is at least as sharp as direct interval evaluation of \( P(x - x_0) \) on \( D \).

We note that in many cases, even tighter bounding of \( B(P) \) is possible.

**Definition 3 (Intrinsic Functions of Taylor Models)** Let \( T = (P, I) \) be a Taylor model of order \( n \) over the \( v \)-dimensional domain \( D = [a, b] \) around the point \( x_0 \). We define intrinsic functions for the Taylor models[1] by performing various manipulations that will allow the computation of Taylor models for the intrinsics from those of the arguments. In the following, let \( f(x) \in P(x - x_0) + I \) be any function in the Taylor model, and let \( c_f = f(x_0) \), and \( \bar{f} \) be defined by \( \bar{f}(x) = f(x) - c_f \). Likewise we define \( P \) by \( P(x - x_0) = P(x - x_0) - c_f \), so that \( (P, I) \) is a Taylor model for \( f \). We proceed the arithmetic of exponential as follows. For the other various intrinsics, refer to [1].

**Exponential.** We first write

\[
 \exp(f(x)) = \exp(c_f + \bar{f}(x)) = \exp(c_f) \cdot \left\{ 1 + \bar{f}(x) + \frac{1}{2!}(\bar{f}(x))^2 + \cdots + \frac{1}{k!}(\bar{f}(x))^k + \frac{1}{(k+1)!}(\bar{f}(x))^{k+1} \exp(\theta \cdot \bar{f}(x)) \right\},
\]

where \( 0 < \theta < 1 \). Taking \( k \geq n \), the part up to \((\bar{f}(x))^n\) is merely a polynomial of \( \bar{f} \), of which we can obtain the Taylor model via Taylor model addition and multiplication. The remaining part will be bounded by an interval. In the actual implementation, one may choose \( k = n \) for simplicity, but it is not a priori clear which value of \( k \) would yield the sharpest enclosures.

**Theorem 1 (Taylor Model Scaling Theorem)** Let \( f, g \in C^{n+1}(D) \) and \( (P_{f,h}, I_{f,h}) \) and \( (P_{g,h}, I_{g,h}) \) be \( n \)th order Taylor models for \( f \) and \( g \) around \( x_h \) on \( x_h + [-h, h]^n \subset D \). Let the remainder bounds \( I_{f,h} \) and \( I_{g,h} \) satisfy \( I_{f,h} = O(h^{n+1}) \) and \( I_{g,h} = O(h^{n+1}) \). Then the Taylor models \( (P_{f+g,h}, I_{f+g,h}) \) and \( (P_{f\cdot g,h}, I_{f\cdot g,h}) \) for the sum and products of \( f \) and \( g \) obtained via addition and multiplication of Taylor models satisfy

\[
 I_{f+g,h} = O(h^{n+1}), \text{ and } I_{f\cdot g,h} = O(h^{n+1}).
\]

Furthermore, let \( s \) be any of the intrinsic functions defined above, then the Taylor model \( (P_{s(f),h}, I_{s(f),h}) \) for \( s(f) \) obtained by the above definition satisfies

\[
 I_{s(f),h} = O(h^{n+1}).
\]

We say the Taylor model arithmetic has the \((n + 1)\)st order scaling property.
The proof for the binary operations follows directly from the definition of the remainder bounds for the binaries. Similarly, the proof for the intrinsics follows because all intrinsics are composed of binary operations as well as an additional interval, the width of which scales at least with the \((n + 1)\)st power of a bound \(B\) of a function that scales at least linearly with \(h\).

**Remark 1 (High Order Scaling Property)**
The high order scaling property of Taylor model arithmetic states that a given function \(f\) can be approximated by another function \(P\) (a polynomial) with an error that scales with high order as the domain decreases. This approximation statement follows standard mathematical practice. However, in the interval community it is customary to study another related but different meaning of scaling: namely the behavior of the overestimation of a given method to determine the range of a function. In the conventional interval community, this scaling property is important because intervals, including range intervals, play a leading role. In the world of Taylor model algorithms, the use of intervals themselves is much reduced, since as a general rule, expressions are kept in Taylor model form as much as possible, for example to retain the ability to suppress dependency. Thus in general, the high order scaling property as stated in the previous theorem is the relevant one. This, however, applies only in a limited sense to the question of range bounding.

Having defined the intrinsics of Taylor model arithmetic as above, we can summarize the main property of Taylor model arithmetic in the following theorem:

**Theorem 2 (Taylor Model Enclosure Theorem)** Let the function \(f : \mathbb{R}^n \to \mathbb{R}^n\) be contained within \(P_f + I_f\) over the domain \(D \subset \mathbb{R}^n\). Let the function \(g : \mathbb{R}^n \to \mathbb{R}\) be given by a code list comprised of finitely many elementary operations and intrinsic functions, and let \(g\) be defined over the range of an enclosure of \(P_f + I_f\). Let \(P + I\) be the result obtained by executing the code list for \(g\) in admissible FP Taylor model arithmetic, beginning with the Taylor model \(P_f + I_f\). Then \(P + I\) is an enclosure for \(g \circ f\) over \(D\).

**Proof.** The proof follows by induction over the code list of \(g\) from the elementary properties of the Taylor model arithmetic. \(\square\)

Apparently the presence of the floating point errors entails that \(P\) is not precisely the Taylor polynomial. In a similar fashion, also the scaling properties of the remainder bound in a rigorous sense is lost. However, these properties of Taylor models are retained in an approximate fashion.

**Remark 2 (Influence of Floating Point Arithmetic)** In the presence of floating point errors, the polynomial \(P\) will be a floating point approximation of the Taylor polynomial of \(g \circ f\) if \(P_f\) was an approximate Taylor polynomial for \(f\). Furthermore, any \((n + 1)\)st order scaling property for the remainder interval will prevail approximately until near the floor of machine precision.

As an immediate consequence, we obtain the next algorithm.

**Algorithm 1 (Range Bounding with Taylor Models)**

**Input:** a finite code list involving elementary operations and intrinsics describing the function \(f\) over the multivariate domain box \(D\)

**Output:** an enclosure of \(f\) in a Taylor model \(P_f + I_f\), and an interval bound \(B(f)\) for the range of \(f\) over \(D\)

1. Set up a Taylor model \(T_I\) enclosing the identity function. This is comprised of the linear multivariate polynomial \(P(x) = x\) plus the remainder bound \([0,0]\).

2. Evaluate the code list for \(f\) in Taylor model arithmetic. As a result, obtain \(P_f + I_f\).

3. Bound the range \(B(P_f)\) of the polynomial \(P_f\), obtain a range bound \(B(f)\) for \(f\) as \(B(f) = B(P_f) + I_f\).
Apparently the sharpness of the range bounding depends on the method to obtain the bound of the polynomial $B(P_f)$. It turns out that in many practical cases, even mere evaluation with intervals yields suitable results that are significantly sharper than what can be obtained with centered and mean value forms. Furthermore, there are various ways to obtain sharper enclosures for $B(P_f)$ that in many cases asymptotically lead to a scaling of the overall error with higher orders.

2 Comparison with Centered and Mean Value Forms

In this section, we compare Taylor models with the centered form (CF) and mean value form (MF) [4, 5] for range bounding for a limited collection of meaningful examples. Over sufficiently small domains, CF and MF usually provide sharper enclosures than intervals, and are known to have the quadratic approximation property. We compare with Taylor model methods of various orders, and subsequent bounding schemes based on either naive interval evaluation of the Taylor polynomial, or based on the linear dominated bounder LDB, an enhancement to the Taylor model bounding that often provides for sharper inclusions. To increase the demand on the LDB method, in all examples shown no domain subdivisions are allowed, even though, apparently, allowing subdivision before applying LDB would increase the applicability of LDB to larger domains. We observe that overall, Taylor models suppress dependency much better than centered forms and mean value forms, resulting in frequently much sharper inclusions. Furthermore, in many cases the LDB method leads to higher order enclosures of estimated ranges.

The computations are performed using COSY for the Taylor models, while intervals, centered forms, and slopes were evaluated using the implementation in the INTLAB toolbox for Matlab[6]. Specifically, we used INTLAB Version 3.1 under Matlab Version 6. Some of the multivariate centered form computations for the normal form problem discussed below took 45 minutes of CPU time, while the Taylor model evaluation of the same function even of order seven could be done in about 20 seconds on the same machine.

We assess the behavior of various algorithms to bound functions with a measure $q$ of relative overestimation[7],

$$ q = \frac{\text{(estimated range)} - \text{(exact range)}}{\text{(exact range)}} $$

We provide logarithmic plots of $q$ as a function of domain width for centered forms (CF), mean value forms (MF), and Taylor models of various orders. Usually, the domain we study has the form $D = x_0 + [-2^{-j}, 2^{-j}]$.

We will also determine empirical approximation orders (EAO) by computing the magnitude of the local slopes of $q$ in a logarithmic plot and adding 1, i.e. \( \text{EAO} = 1 + \frac{d \log(q)}{d \log(|D|)} \). With this definition, the interval evaluation will commonly have EAO of 1, while centered forms and mean value forms will have order 2. We usually list the EAO only until the floor of machine precision is reached.

For notational simplicity, in the following pictures, results obtained using interval evaluation will be denoted by the symbol $\Box$, reminiscent of an interval box, while those obtained by the mean value form and centered form will be denoted by the symbols $\nabla$ and $\triangle$, reminiscent of a gradient and a difference quotient, respectively. Taylor models will be identified by numbers corresponding to their orders.

We begin our discussion with the study of a simple three dimensional example function with modest dependency but overall rather innocent behavior studied in [1]. The function has the form

$$ f_1(x, y, z) = \frac{4 \tan(3y)}{3x + x \sqrt{-7(x-8)} - 120 - 2x - 7z(1+2y) - \sinh \left(0.5 + \frac{6y}{8y+7} \right) + \frac{(3y + 13)^2}{3z} - 20z(2z - 5) + \frac{5x \tanh(0.9z)}{\sqrt{5y}} - 20y \sin(3z), $$

and the function is defined for $0 < x < 8$, $y > 0$, and $z \neq 0$. We study the behavior on the domain interval boxes $(2, 1, 1) + [-2^{-j}, 2^{-j}]^3$ and show the results in figure 1. As a function of $j$, we show
log_{10}(q)$ for interval evaluation, centered and mean value form as well as TM range bounding by mere interval evaluation of the Taylor polynomial, and TM range bounding through LDB of orders 3, 6, and 9. We also plot the EAO for both of these cases.

It can be seen that all Taylor model methods achieve enclosures that are significantly sharper than CF and MF, showing the ability of the Taylor model method to suppress whatever dependency there is in the function. Without LDB, the approximation order of CF, MF and all TM methods is 2. CF uniformly provides slightly sharper enclosures as MF, as is frequently observed. The first order Taylor model method behaves similar to CF, and is in fact slightly superior. The higher order Taylor models, while still showing order 2 scaling, provide enclosures that is about 1 order of magnitude sharper than those of CF.

In order to study the behavior of the suppression of dependency in more detail, let us study in the same domain the following function

$$f_2(x, y, z) = f_1 + \sum_{j=1}^{10} (f_1 - f_1)$$

which is obtained by repeatedly adding and subtracting the function, such that for the actual function values we have $f_2(x, y, z) = f_1(x, y, z)$, but the code list for $f_2$ exhibits a more pronounced cancellation problem. The results are shown in figure 1.

Overall the behavior of the methods is similar to before; however, we observe that now, the non-LDB Taylor model methods of orders 6 and 9 uniformly provide a sharpness of enclosure that is around 2 orders of magnitude better than those of CF; The third order Taylor model reaches this level only at $j = 4$. This difference in sharpness is 10 times greater than in the previous example. Apparently the TM method is affected very little by the fact that the function is added and subtracted from itself 10 times. In fact, direct comparison of the TM curves shows that the actual overestimation is very nearly the same as in the previous example, while it increases by a factor of 10 for CF, MF, and first order Taylor models. With LDB, the approximation order of the Taylor model of order $n$ increases to $(n + 1)$, until the floor of the machine precision is reached. At the most favorable point, the sharpness of the 9th order Taylor model method is about 11 orders of magnitude higher than that of CF.

As another challenging example we study a normal form defect function, an example of the class of functions that originally led to the development of the Taylor model methods. The function has the form

$$f_4(x_1, \ldots, x_6) = \sum_{i=1}^{3} \left( \sqrt{y_{2i-1}^2 + y_{2i}^2} - \sqrt{x_{2i-1}^2 + x_{2i}^2} \right)^2$$

where $\tilde{y} = \tilde{P}_1 \left( \tilde{P}_2 \left( \tilde{P}_3(\tilde{x}) \right) \right)$

and $\tilde{P}_1$, $\tilde{P}_2$ and $\tilde{P}_3$ are six-dimensional vectors of polynomials in six variables of degrees ranging from around 5 to around 10. For our purposes, the relevant properties of the function is that it has function values very near to zero, while each of the polynomials $\tilde{P}_1$, $\tilde{P}_2$ and $\tilde{P}_3$ can exhibit large coefficients. Since the polynomials themselves have several thousand terms, there is thus a very pronounced dependency problem. Furthermore, the dependency problem increases more and more for larger values of the arguments, and so the functions offer a convenient way to study the behavior of bounding tools at various levels of dependency. In the examples of our calculation, the polynomials $\tilde{P}_1$, $\tilde{P}_2$ and $\tilde{P}_3$ are of degree 5, and they are available at [8]; in this case, the degree of the function $f_4(x_1, \ldots, x_6)$ is 250.

We again compare the performance of Taylor models with CF, MF, and intervals. For technical reasons connected to the evaluation of the polynomials in COSY, the order of computation had to be chosen at least as high as that of the polynomials $\tilde{P}_1$, and we picked orders 5, 6 and 7. In figure 2, we show the results for the domains $D = 0.1 \cdot (1 + [-2^{-j}, 2^{-j}])^6$. The LDB enhanced TM method begins to improve the accuracy from $j = 3$. For $j = 7$, the TM method of order 7 outperforms CF by around 14 orders of magnitude, while the TM of order 5 outperforms CF by around 8 orders of magnitude. As the plots of EAO shows,
the LDB TM of order \( n \) achieves orders \( (n + 1) \) as expected.

3 Comparison with Remainder Bounds from Interval AD

The use of automatic differentiation (AD) methods [2] for the computation of accurate derivatives from code lists has a history nearly as long as that of interval analysis itself[4, 5]. In the interval framework, the method can be used to provide enclosures for derivatives by merely executing AD code with interval coefficients, where the initial interval has to enclose the domain of interest for the derivatives. In our context, this interval automatic differentiation (IAD) method allows to compute remainder bounds of functions by using Taylor’s remainder formula, and rigorously bounding the high-order partial derivatives that appear in the remainder term.

A practical inconvenience of this approach is that one has to perform two independent executions of the code list, one with narrow intervals to obtain the Taylor coefficients, and another one with wide intervals to obtain the remainder bound. However, the major limitation of the method is that, different from the Taylor model approach which can often alleviate the dependency problem of a given function, this approach cannot alleviate dependency, but frequently even has the tendency to enhance the dependency problem.

The reason for this behavior lies in the fact that the actual code list for the derivative computation, which is evaluated with wide intervals making it susceptible to dependency, contains all parts of the code of the function, plus the additional code necessary to propagate derivatives. The length of the resulting code list, and hence the potential for overestimation, apparently increases with both order and dimensionality, and so the IAD method is thus expected to suffer more and more just in the terrain where the Taylor model method becomes better and better. Besides, of course we also expect that the performance of IAD suffers more if the code list itself becomes longer, just as any other interval evaluation. On the other hand, in the case of the Taylor model computation, the new contributions to the remainder bounds are always computed from the Taylor expansion of the current intermediate variables in the code list, which is not subject to dependency.

In order to study a realistic and demanding example, we investigate the normal form defect function \( f_4 \) in eq. (1). We look at the remainder bounds calculated by IAD and TMs of order 5, 6 and 7. The left picture in figure 3 shows the actual magnitude of the remainder bounds calculated by both methods for the domain \( (0.2 + [-2^i, 2^i])^6 \) for various values of \( j \). We see that due to the enormous complexity of the function, both methods have large overestimation for \( j = 1 \). Around \( j = 3 \), the remainder bounds calculated with TM fall below 1, while at this point, those calculated by IAD are still near \( 10^{30} \). It is important to note that in the TM calculations, the remainder bounds also absorb the errors of the polynomial coefficient arithmetic, which ultimately puts a lower limit on their size. On the other hand, in the case of the IAD computation, these terms are not included because the polynomial part is computed separately and not even known in our computation; so for small domains and sharp enclosures, the IAD results are expected to be overly optimistic. The right picture shows the ratio between the IAD remainder bound and those obtained by TMs for orders 5 and 7. The ratio ranges from \( 10^{42} \) to about \( 10^8 \), and as expected, for higher orders, TMs show a more favorable behavior.

Altogether it is apparent that while IAD can be used to obtain remainder bounds with the high-order scaling property, the dependency of the examples makes the TM remainders overall much more favorable. For computations of limited length and limited dependency, this may be of minor significance, but its effects will become more and more dramatic for more complicated functional dependencies.

4 Conclusion

The main aspects of the Taylor model (TM) method have been reviewed. The method is then compared to a variety of other state of the art tools. When compared with the centered forms
(CF) and mean value form (MF) for purposes of range bounding, it is found that first order TMs behave similar to CF which in turn behaves generally better than MF, although the first order TMs seem to have a tendency to outperform CF by a slight margin. However, higher order TMs are found to suppress the dependency problem significantly better than either CF and MF. It is shown that the remainder bound of the TM of order \( n \) scales with order \( (n + 1) \), and this behavior is also observed in practical computations. This enables to provide validated approximations of complicated functional dependencies with an accuracy that scales with a high order of the domain width. When TMs are combined with advanced bounders for the polynomial part such as the linear dominated bounder LDB or other tools[9], higher order range bounding can be achieved.

The interval automatic differentiation (IAD) method can also be used to obtain bounds for the remainder of a Taylor expansion. However, different from the TM approach, this method suffers from a dependency problem that is usually significantly worse than that of the original function. As a consequence, the practical performance is often significantly affected, and in general for sufficiently complicated functions, the sharpness of the resulting remainder bounds cannot come close to those that can be obtained via TMs.

Acknowledgements
We gratefully acknowledge financial support through the US Department of Energy, Grant Number DE-FG02-95ER4093, the National Science Foundation, the German National Merit Foundation, the Illinois Consortium for Accelerator Research, and an Alfred P. Sloan Fellowship.

References:
Figure 1: Relative overestimation \( q \) and EAO (empirical approximation orders) for the function \( f_1(x, y, z) \) (top) and \( f_2(x, y, z) \) (middle) in the domain \((2, 1, 1) + [\!-2^{-j}, 2^{-j}]\) with naive interval range bounding, and \( f_2 \) in the same domain with LDB range bounder (bottom).
Figure 2: $q$ and EAO for the 6D normal form deviation function $f_4(\bar{x})$ in the domain $0.1 \cdot (1 + [-2^{-j}, 2^{-j}])^6$.

Figure 3: Remainder intervals of the normal form function $f_4$. Left: Width. Right: Ratio of the widths of remainders obtained by IAD and TM. $I_{TM}$ includes the bounds for the floating point error of the polynomial coefficient part, which is not included in $I_{IAD}$.