Improved Validated Bounds for Taylor Coefficients and for Taylor Remainder Series†

M. Neher, Karlsruhe

Abstract

This paper presents methods for the validated computation of bounds for Taylor coefficients and bounds for Taylor remainder series of analytic functions. These bounds are derived from modifications of Cauchy's estimate.

The proposed methods have been implemented in mathematical software called ACETAF. Interval arithmetic is used to restore mathematical rigour to practical calculations. The performance of ACETAF is demonstrated with numerical examples.

MSC Subject Classifications: 65G20, 65G30, 30B10, 30-04.

Key Words: Bounds for Taylor coefficients of analytic functions, complex interval arithmetic, enclosure methods, mathematical software.

1 Introduction

This paper is concerned with the practical calculation of bounds for Taylor coefficients. For a given analytic function f, we construct bounds that form a geometric series or some antiderivative of a geometric series. Summation of this series yields a bound for the Taylor remainder series of f.

Such bounds are used for the error analysis in the well–known Taylor series method for the solution of ODEs. For example, consider the scalar IVP

$$\begin{aligned} y^{(n)} &= \sum_{i=0}^{n-2} p_i(x)\, y^{(i)} + p_{-1}(x), \quad x \in (-r,r),\ r>0, \\ y^{(i)}(0) &= y_{io}, \quad i = 0,\dots,n-1, \end{aligned} \tag{1}$$

where the functions p_i are assumed to be analytic in $(-r,r)$ having series expansions

$$p_i(x) = \sum_{j=0}^{\infty} b_{ij}\, x^j, \quad x \in (-r,r), \quad i = -1,\dots,n-2. \tag{2}$$

†Appeared in: J. Comput. Appl. Math. **152** (2003), 393-404

The solution of (1) can be written as a power series

$$y(x) \ := \ \sum_{\nu=0}^{\infty} a_\nu \, x^\nu, \quad x \in (-r, r). \tag{3}$$

In the Taylor series method, a finite number of coefficients a_ν of (1) are obtained from recurrence relations, and the Taylor polynomial

$$\tilde{y}(x) \ := \ \sum_{\nu=0}^{k-1} a_\nu \, x^\nu \quad \text{for some } k \in \mathbb{N}$$

serves as an approximate solution of (1). If suitable bounds for the Taylor coefficients b_{ij} of the analytic functions $p_i(x)$ in (2) are available then the approximation error $|y - \tilde{y}|$ can be estimated as follows:

Theorem [Neher 2001b]

(i) If there are constants $m_i \in \mathbb{N}_0$ and $B_i \geq 0$ such that

$$|b_{ij}| \ \leq \ \frac{B_i}{r^j}, \quad \text{for } j > m_i, \quad i = -1, \ldots, n-2 \tag{4}$$

then there exist numbers $\kappa \in \mathbb{N}$ and $A > 0$ such that

$$|a_k \, r^k| \ \leq \ A \ := \ \max_{\nu=0}^{\kappa} |a_\nu \, r^\nu| \quad \text{for all} \ \ k \in \mathbb{N}_0 \, . \tag{5}$$

(ii) Under the above assumptions, for $x \in (-r, r)$ and all $k \in \mathbb{N}$,

$$|\, y(x) - \sum_{\nu=0}^{k-1} a_\nu \, x^\nu \,| \ \leq \ \frac{A}{r^{k-1}} \ \cdot \ \frac{x^k}{r - x}.$$

An algorithm for the practical computation of κ and A from given data m_i, B_i in (4) is discussed in [Neher 2001b]. Nonlinear examples for this error analysis for ODEs are presented in [Neher 1997].

Geometric series bounds for Taylor coefficients of analytic functions are also used for the determination of multiple zeros or clusters of zeros. In [Sakurai and Sugiura 2000] the availability of bounds according to (4) was assumed, but no method for their computation was mentioned.

This paper addresses the computation of bounds for the Taylor coefficients of a given analytic function f. Its theoretical foundation has already been developed in [Neher 2001c]. Here, we present improved algorithms for the practical calculation of the bounds. The validated estimation of the Taylor remainder series is considered for the first time in this paper.

2 Estimates for Taylor Coefficients

In the following, let $f(z) = \sum_{j=0}^{\infty} a_j z^j$ be analytic in B and bounded on C, where B is the complex disc $\{z : |z| < r\}$ and C the circle $\{z : |z| = r\}$, for some $r > 0$. A well known bound for the Taylor coefficients of f is Cauchy's estimate $M(r)$ (cf. [Henrici 1974, p. 84]):

$$|a_j| \;\leq\; \frac{M(r)}{r^j}, \quad M(r) := \max_{|z|=r} |f(z)|, \quad j \in \mathbb{N}_0.$$

Unfortunately, Cauchy's estimate is sometimes very pessimistic. To obtain better bounds, two modifications of Cauchy's estimate were proposed in [Neher 2001c]. The first uses Taylor polynomial approximations to f, the second uses derivatives:

Theorem 1 *Let f be analytic in B and bounded on C. Furthermore, let $t_l(z)$ denote the Taylor polynomial of order l to f. Then*

$$|a_j| \;\leq\; \frac{N(r,l)}{r^j} \quad \textit{for } j > l, \quad \textit{where} \quad N(r,l) := \max_{|z|=r} |f(z) - t_l(z)|\,.$$

Theorem 2 *Let f be analytic in B and let the m–th derivative of f be bounded on C. Then*

$$|a_j| \;\leq\; \frac{U(r,m)r^m}{P(j-m,m)r^j} \quad \textit{for } j \geq m, \tag{6}$$

where

$$U(r,m) := \max_{|z|=r} |f^{(m)}(z)|\,, \quad P(j,m) := \frac{(j+m)!}{j!}.$$

It was mentioned in [Neher 2001c] that the last two methods could be combined. Let $\widehat{t_l}$ be the Taylor polynomial of order l for $f^{(m)}$. Then instead of (6), we have

$$|a_j| \;\leq\; \frac{V(r,m,l)r^m}{P(j-m,m)r^j} \quad \text{for } j > m+l,$$

where

$$V(r,m,l) := \max_{|z|=r} |f^{(m)}(z) - \widehat{t_l}(z)|\,.$$

However, this estimate has not been found useful in practical calculations. It has been experienced in many numerical examples that $U(r, m+l)$ yielded better bounds than $V(r,m,l)$, after less computation time.

3 Estimates for Taylor Remainder Series

The above estimates for the Taylor coefficients are the basis of the estimation of the remainder series $R_p(z) := \sum_{j=p+1}^{\infty} a_j z^j$. Here, we are looking for a bound on $R_p(z)$ at some point z with $|z| = \omega r$, $\omega \in (0,1)$.

Using Cauchy's estimate, we get

$$|R_p(z)| \;\le\; \sum_{j=p+1}^{\infty} M(r)\,\omega^j \;=\; M(r)\,\frac{\omega^{p+1}}{1-\omega} \quad \text{for } p \ge 0. \tag{7}$$

Following Theorem 1, we have

$$|R_p(z)| \;\le\; N(r,l)\,\frac{\omega^{p+1}}{1-\omega} \quad \text{for } p \ge l, \tag{8}$$

whereas Theorem 2 yields

$$|R_p(z)| \;\le\; U(r,m)\,r^m \sum_{j=p+1}^{\infty} \frac{\omega^j}{P(j-m,m)} \quad \text{for } p \ge m-1 \ge 0.$$

For $p = m-1$, the summation can be made explicit. The proof of the following Theorem (which appears to be new) is given in the appendix.

Theorem 3 *For $m \in \mathbb{N}$ and $\omega \in (0,1)^{\dagger}$,*

$$\sum_{j=m}^{\infty} \frac{\omega^j}{P(j-m,m)} \;=\; (\omega-1)^{m-1}\left(z_{m-1} - \frac{\ln(1-\omega)}{(m-1)!}\right) - \sum_{j=0}^{m-2} \frac{(-1)^{m-1-j}}{j!} z_{m-1-j}\,\omega^j, \tag{9}$$

where the numbers z_j are defined by the recursion

$$z_0 := 0, \quad z_j := \frac{1}{j}\left(z_{j-1} + \frac{1}{j!}\right), \quad j = 1, 2, \ldots. \tag{10}$$

Estimates for Taylor remainder series of arbitrary index follow from (9). For $p \ge m-1$ we get

$$\begin{aligned} |R_p(z)| \;\le\; U(r,m) r^m \Bigg\{ & (\omega-1)^{m-1}\left(z_{m-1} - \frac{\ln(1-\omega)}{(m-1)!}\right) \\ & - \sum_{j=0}^{m-2} \frac{(-1)^{m-1-j}}{j!} z_{m-1-j}\,\omega^j - \sum_{j=m}^{p} \frac{\omega^j}{P(j-m,m)} \Bigg\}. \end{aligned} \tag{11}$$

†The assertion holds for $\omega \in [-1,1)$, but only $\omega > 0$ is required in this paper.

4 Implementation

When the above estimates are implemented in a computer program, the major obstacle for exactness in computation is the finite arithmetic on digital computers. Even if roundoff errors are small, they still falsify the result of a practical calculation.

Floating point interval arithmetic [Kulisch and Miranker 1981] has been found a convenient tool to restore mathematical rigour in numerical computations. In floating point interval arithmetic, all calculations are performed on intervals with machine representable bounds instead of floating point numbers, and executed according to the rules of interval arithmetic [Moore 1966; Alefeld and Herzberger 1983; Neumaier 1990; Jaulin et al. 2001]. The exact result of any arithmetic operation is automatically enclosed in a floating point interval, including roundoff errors.

Using floating point interval arithmetic, the validated computation of the estimates on Taylor coefficients and Taylor remainder series has been implemented in a computer program called ACETAF. With ACETAF, it is possible to compute bounds for the Taylor coefficients of analytic compositions of rational functions and of the usual complex standard functions (like e^z, $\sin z$, $\operatorname{Log} z$, $\ldots$).

ACETAF contains a complex function library that is based on the algorithms discussed in [Braune and Krämer 1987; Bühler 1993], which provide the best possible interval bounds for the ranges of the respective functions. Besides these range enclosures, the second important tool for the validated computation of ranges of concatenated complex functions is the complex mean value form which was developed in [Neher 2001a]:

Theorem 4 *Let f be analytic in a domain $D \subseteq \mathbb{C}$, let $Z \subseteq D$ be a rectangular complex interval and let $z_0 = x_0 + \imath y_0$ be a point in Z. Furthermore, let $F'(Z)$ denote a rectangular complex interval that encloses the range of f' on Z. Then the following inclusion holds for the range $f(Z)$:*

$$f(Z) \subseteq f(z_0) + F'(Z)(Z - z_0). \tag{12}$$

While direct interval evaluation of a concatenated function usually converges linearly, the mean value form converges quadratically in the sense that the range overestimation is proportional to the square of the diameter of the argument interval.

Derivatives, which are required in ACETAF for the mean value form and for the computation of $N(r, l)$ and $U(r, m)$, are calculated with automatic differentiation [Rall 1981; Griewank 2000]. In our implementation, the automatic computation of $F'(Z)$ requires the computation of $F(Z)$ first, which is performed by direct interval evaluation of the inclusion function F. In this case, the intersection of $F(Z)$ with the mean value form is an effective means to improve the range enclosure at negligible costs.

Well known branch and bound methods for rigorous global optimization [Ratschek and Rokne 1988; Hansen 1992; Kearfott 1996] are employed in the practical calculation of $M(r)$, $N(r,l)$, or $U(r,m)$. We now comment on the respective methods for the validated computation of these estimates.

4.1 Validated computation of $M(r)$

The computation of $M(r)$ via a global optimization problem for $|f|$ on C has already been described in [Neher 2001c]. The discussion is summarized here for clarity.

To calculate a validated upper bound for $M(r)$, the circle C is covered with complex intervals Z_k, $k = 1, \ldots, k_{\max}$, which are gathered in a list L. From each Z_k, a particular number c_k is chosen. Using complex interval functions, the function value enclosures

$$[\underline{f}_k, \overline{f}_k] \supseteq |f(c_k)| \text{ for all } k$$

and range enclosures

$$[\underline{F}_k, \overline{F}_k] \supseteq |f(Z_k)| \text{ for all } k$$

are computed. We then have

$$\underline{M} := \max_k \underline{f}_k \leq M(r) \leq \max_k \overline{F}_k =: \overline{M}. \tag{13}$$

If the diameter of the interval $[\underline{M}, \overline{M}]$ is large then the bounds are refined iteratively. Intervals Z_k for which $\overline{F}_k < \underline{M}$ holds cannot contain an extremal point. These intervals are removed from the list L. Subdividing the remaining intervals and evaluating the function values again, new bounds $\underline{f}_k$, $\overline{F}_k$ are obtained, from which improved bounds $\underline{M}$ and $\overline{M}$ follow. This process is being continued until $M(r)$ is determined with sufficient accuracy. The success of this method lies in the fact that if $|f|$ does not have too many global maxima then usually many intervals Z_k can be removed from the list L after each subdivision step, and accurate bounds for $M(r)$ are obtained with only a few function values.

Here, it is often sufficient to use natural interval extensions of f. The mean value form can improve the accuracy of the computed bound for $M(r)$ if the evaluation of f involves many operations, but it will take more computation time because it requires values of derivatives.

4.2 Validated computation of $N(r,l)$

The same optimization algorithm was used in [Neher 2001c] to compute $N(r,l)$. However, it appeared that this method was not optimal, and that two important modifications were necessary to make it more accurate and effective in practice.

Computing $N(r,l)$ instead of $M(r)$ only makes sense if t_l is a good approximation to f. In this case, however, severe cancellation occurs in the computation of $f(Z_k) - t_l(Z_k)$. In interval arithmetic, this cancellation causes a large overestimation of the number $N(r,l)$, because we have

$$w(f(Z_k) - t_l(Z_k)) = w(f(Z_k)) + w(t_l(Z_k))$$

and $w(f(Z_k)) \gg 0$ in practical problems.

The prevent such an overestimation, the complex mean value form

$$(f - t_l)(Z_k) \subseteq f(c_k) - t_l(c_k) + (f'(Z_k) - t_l'(Z_k))(Z_k - c_k).$$

should be used. Here, the cancellation occurs in the subtraction of two floating point numbers instead of intervals, where it is less harmful. Compared to earlier calculations, with the introduction of the mean value form the bounds on $N(r,l)$ were improved by several orders of magnitude.

A second modification is necessary to make the optimization procedure effective. The l–th order best approximation polynomial in the maximum norm attains its maximum distance from f at least $l+2$ times on C [Walsh 1935, p. 21]. The Taylor polynomial, a near–best approximation [Geddes and Mason 1975], exhibits a similar behaviour. In many numerical examples it was observed that the distance from f of a higher order Taylor polynomial attains many near–global local maxima on C and that in the early stages of the optimization procedure, when the diameters of the intervals Z_k were still large, $|f - t_l|$ was uniformly small on C compared to the widths of the interval arithmetic evaluation of $|f - t_l|$. Unless the diameters of the Z_k became sufficiently small, no intervals were removed from the list L, so that all function evaluations in the initial subdivision steps were obsolete.

To save computation time, the following method has been found useful in practice: instead of choosing a constant order l of the Taylor polynomial, we fix the maximum number $k_{\max}$ of intervals that are used in the GOP algorithm. The optimal order l_{opt} is then computed as the number for which a partition of C with k_{max} segments of equal size yields the smallest bound for $N(r,l)$. A strategy for the determination of l_{opt} has been described in [Eble and Neher 2001].

4.3 Validated computation of $U(r, m)$

Like $|f|$, $|f^{(m)}|$ usually has only a few global maxima, and the adaptive optimization algorithm works well. If m is large then the intersection of $f^{(m)}(Z)$ with the mean value form

$$f^{(m)}(c) + f^{(m+1)}(Z)(Z - c)$$

is about twice as expensive as the evaluation of $f^{(m)}(Z)$, but the improvement of the range enclosure is often worth the additional effort.

The bounds on the Taylor coefficients that result from $U(r, m)$ are sometimes several orders of magnitude better than the bounds that result from $M(r)$ or $N(r, l)$. On the other hand, computing some higher order derivative of a complicated function with automatic differentiation can be quite expensive. For large values of r and m, the computation times for the calculation of $U(r, m)$ can get very large.

4.4 Validated computation of R_p

The interval evaluation of the estimates (7) or (8) is straightforward. However, this does not hold for the interval evaluation of (11), which suffers from severe cancellation for $p \gg m - 1$. To see this, let $b_j := \omega^j / P(j - m, m)$ and let $S_p := \sum_{j=p+1}^{\infty} b_j$. Then in (11), S_p is computed as

$$S_p \;=\; S_{m-1} - \sum_{j=m}^{p} b_j. \tag{14}$$

Because $\{b_j\}_{j=m}^{\infty}$ is a rapidly converging series, we have $S_p \approx b_{p+1}$ and $S_p \gg S_{p+1}$, so that cancellation in (14) is inevitable.

On the other hand, the numbers b_j are upper bounds to the Taylor coefficients. Even the exact value of R_p already overestimates the remainder series. Hence, a slight additional overestimation isn't critical. Because $P(j - m, m)$ is a monotonously increasing function of j, a validated upper bound to S_p is

$$\frac{1}{P(p+1-m, m)} \sum_{j=p+1}^{\infty} \omega^j \;=\; \frac{1}{P(p+1-m, m)} \frac{\omega^{p+1}}{1-\omega},$$

which yields

$$|R_p(z)| \;\le\; \frac{U(r, m)\, r^m}{P(p+1-m, m)} \frac{\omega^{p+1}}{1-\omega}. \tag{15}$$

(15) is evaluated without cancellation, and it has been found of reasonable accuracy in many numerical examples.

4.5 Distribution of ACETAF

ACETAF has been written in C++. The program is available in two versions, depending on the interval library that is used: C–XSC [Klatte et al. 1993] or filib++ [Lerch et al. 2001a; Lerch et al. 2001b]. The C–XSC library is more comprehensive than filib++, but the latter is much faster. Users who want to use ACETAF as a stand-alone program should use the filib++ version. Those who wish to integrate ACETAF in their existing C–XSC programs will require the C–XSC version. The software is available at the following sites:

C–XSC and filib++: `http://www.xsc.de`

ACETAF: `http://www.uni-karlsruhe.de/~Markus.Neher/acetaf.html`

5 Numerical Examples

The following numerical examples were computed with ACETAF 2.71. For each example, we show a table of upper bounds for $M(r)$, $N(r,l)$, $U(r,m)$ and R_p, for several radii. The tables include bounds for some of the Taylor coefficients of the respective functions and the computation times (in seconds) for the filib++ interval library on a PC with a 1200 MHz Athlon processor. With the C–XSC interval library, identical results are obtained, but the computation times are about ten times as large.

For two examples, we compare our results with bounds that were published in [Neher 2001c], to demonstrate the improvement due to the mean value form in the computation of $N(r,l)$.

Example 1: Bounds for the Taylor Coefficients of e^z.

Table 1 shows the performance of the various methods for the exponential function. M, N and U are all computed very fast, but this is in part due to the simplicity of the higher order derivatives. As can be observed, N is smaller than M by several powers of ten, and U yields much better bounds for the Taylor coefficients and the remainder series of f.

In Table 2, we show the improvement due to the mean value form for the computation of $N(r,l)$. The maximal number $k_{\max}$ of subintervals that were used in the computation is also given. The orders of the Taylor polynomials were chosen automatically by the respective program versions. The underlying heuristics [Eble and Neher 2001] have been found reliable and almost optimal (with respect to the accuracy of $N(r,l)$) in many numerical examples. The linear convergence of the direct inclusion functions in ACETAF 1.0 and the quadratic convergence of the mean value form in ACETAF 2.71 are both well observed.

r	l/m	$M/N/U$	a_{100}	a_{1000}	$R_{49}(0.95r)$	$R_{99}(0.95r)$	Time
1	—	M=2.8E+00	2.8E+00	2.8E+00	4.4E+00	3.4E−01	< 1
1	l=10	N=1.3E−06	1.3E−06	1.3E−06	2.0E−06	1.6E−07	< 1
1	m=50	U=2.8E+00	9.2E−94	9.8E−150	2.7E−66	5.2E−95	< 1
10	—	M=2.7E+04	2.7E−96	2.7E−996	4.1E+04	3.2E+03	< 1
10	l=28	N=7.5E−01	7.5E−101	7.5E−1001	8.9E−02	5.3E−04	1.4
10	m=50	U=2.7E+04	8.6E−140	9.2E−1096	2.5E−12	4.9E−41	< 1
20	—	M=5.9E+08	4.7E−122	5.5E−1293	9.0E+08	7.0E+07	< 1
20	l=42	N=6.5E+04	5.1E−126	6.0E−1297	9.9E+04	7.7E+03	1.9
20	m=50	U=5.9E+08	1.7E−150	2.2E−1377	6.3E+07	1.3E−21	< 1

Table 1: Bounds for $f(z) = e^z$.

r	$k_{\max}$	ACETAF 1.0		ACETAF 2.71	
		l	N	l	N
1	8192	8	1.7E−03	8	1.3E−06
1	32768	8	4.3E−04	11	8.2E−08
10	8192	24	1.6E+02	28	7.5E−01
10	32768	25	4.1E+01	30	4.8E−02
20	8192	38	7.3E+06	42	6.5E+04
20	32768	40	1.9E+06	46	4.0E+03

Table 2: Bounds for $N(r, l)$ for $f(z) = e^z$.

Example 2: Bounds for the Taylor Coefficients of $\tanh(\ln(z + 11)/3)$.

The function of this example has a singularity at $z = -11$, and the absolute values of the derivatives of f grow strongly near that point. If r is large then $M(r)$ and $N(r, l)$ give better bounds for the Taylor coefficients a_j of f with small indexes j than does $U(r, m)$. Only for large indexes j, $U(r, m)$ has the advantage due to the better asymptotic behaviour for $j \to \infty$.

For $r = 10$, the order l of the Taylor polynomial must be high for a good approximation. A tight interval arithmetic evaluation of $f - t_l$ becomes difficult, but nevertheless there is a decisive improvement of $M(10)$ by $N(10, 54)$.

r	l/m	$M/N/U$	a_{100}	a_{1000}	$R_{49}(0.95r)$	$R_{99}(0.95r)$	Time
1	—	$M = 6.8\text{E}{-01}$	6.8E−01	6.8E−01	1.1E+00	8.1E−02	< 1
1	$l = 8$	$N = 1.3\text{E}{-09}$	1.3E−09	1.3E−09	2.0E−09	1.6E−10	2.2
1	$m = 50$	$U = 1.6\text{E}{+12}$	5.1E−82	5.4E−138	1.5E−54	2.9E−83	3.0
5	—	$M = 7.3\text{E}{-01}$	9.3E−71	7.9E−700	1.2E+00	8.7E−02	< 1
5	$l = 20$	$N = 1.1\text{E}{-07}$	1.3E−77	1.1E−706	1.6E−07	1.3E−08	2.7
5	$m = 50$	$U = 1.4\text{E}{+23}$	5.2E−106	4.7E−791	1.2E−08	2.3E−37	26
10	—	$M = 7.8\text{E}{-01}$	7.8E−101	7.8E−1001	1.2E+00	9.2E−02	< 1
10	$l = 54$	$N = 2.0\text{E}{-04}$	2.0E−104	2.0E−1004	—	2.3E−05	3.9
10	$m = 50$	$U = 3.0\text{E}{+62}$	9.6E−82	1.1E−1037	2.8E+46	5.5E+17	53

Table 3: Bounds for $f(z) = \tanh(\ln(z + 11)/3)$.

Example 3: Bounds for the Taylor Coefficients of $(\cos z)/(z^2 + 101)$.

r	l/m	$M/N/U$	a_{100}	a_{1000}	$R_{49}(0.95r)$	$R_{99}(0.95r)$	Time
1	—	$M = 1.6\text{E}{-02}$	1.6E−02	1.6E−02	2.5E−02	1.9E−03	< 1
1	$l = 10$	$N = 7.2\text{E}{-09}$	7.2E−09	7.2E−09	1.1E−08	8.5E−10	1.8
1	$m = 50$	$U = 3.4\text{E}{+18}$	1.1E−75	1.2E−131	3.2E−48	6.2E−77	1.8
5	—	$M = 1.2\text{E}{+00}$	1.5E−70	1.2E−699	1.8E+00	1.4E−01	< 1
5	$l = 26$	$N = 1.3\text{E}{-05}$	1.7E−75	1.4E−704	2.0E−05	1.5E−06	2.4
5	$m = 50$	$U = 7.6\text{E}{+31}$	2.8E−97	2.5E−782	6.3E+00	1.3E−28	50
10	—	$M = 1.3\text{E}{+04}$	1.3E−96	1.3E−996	4.2E+04	3.2E+03	< 1
10	$l = 51$	$N = 9.5\text{E}{+03}$	9.5E−97	9.5E−997	—	1.2E+03	3.2
10	$m = 50$	$U = 2.3\text{E}{+134}$	7.5E−10	8.0E−966	2.2E+118	4.3E+89	49

Table 4: Bounds for $f(z) = (\cos z)/(z^2 + 101)$.

f has a singularity at $z = \sqrt{101}i$, and the circle with radius 10 comes very close to this point. The computation of M and U still works, but $U(10, m)$ rapidly increases with m. Also, the computation times for $U(r, m)$ become large when m becomes large, because the evaluation of higher order derivatives is expensive.

In Table 5, we compare direct inclusion functions and the mean value form for the computation of $N(r, l)$. For small radii, there is a strong improvement of the bounds by the mean value form. For r=10, neither method is competitive because the optimal orders that are required for a good approximation of f are so large that overestimations in the interval arithmetic function evaluations prevent cost–effective improvements of Cauchy's estimate $M(10)$.

r	$k_{\max}$	ACETAF 1.0		ACETAF 2.71	
		l	N	l	N
1	8192	8	6.4E−06	10	7.2E−09
1	32768	8	1.6E−06	13	4.4E−10
5	8192	20	4.0E−03	26	1.3E−05
5	32768	22	1.0E−03	30	8.0E−07
10	8192	50	1.2E+04	51	9.5E+03
10	32768	50	1.2E+04	66	8.3E+03

Table 5: Bounds for $N(r, l)$ for $f(z) = (\cos z)/(z^2 + 101)$.

Conclusion

We have presented several methods for the practical calculation of validated bounds for Taylor coefficients of analytic functions. The applicability of these methods has been demonstrated with numerical examples.

Future work will concentrate on integrating the estimates into software for the validated solution of ODEs.

Appendix

Proof of Theorem 3

For $m \in \mathbb{N}_0$, let $g_m(\omega) := \sum_{j=m}^{\infty} \frac{\omega^j}{P(j-m,m)}$. Then we have

$$\begin{aligned} g_m'(\omega) &= \sum_{j=m}^{\infty} \frac{j\,\omega^{j-1}}{P(j-m,m)} = \sum_{j=m}^{\infty} \frac{\omega^{j-1}}{P(j-m,m-1)} \\ &= \sum_{j=m}^{\infty} \frac{\omega^{j-1}}{P(j-1-(m-1),m-1)} = \sum_{j=m-1}^{\infty} \frac{\omega^{j}}{P(j-(m-1),m-1)} \\ &= g_{m-1}(\omega) \end{aligned}$$

and

$$g_0(\omega) = \sum_{j=0}^{\infty} \omega^j = \frac{1}{1-\omega}.$$

Hence, $g_m(\omega)$ is obtained by repeated integration of $\frac{1}{1-\omega}$. Because $g_m(0) = 0$ holds for all $m \geq 1$, we have

$$g_{m+1}(\omega) = \int_0^{\omega} g_m(t)dt \quad \text{for } m = 0, 1, \ldots, . \tag{16}$$

The assertion of Theorem 3 now follows by induction. It is obviously true for $m = 1$. Now suppose that (9) holds for some $m \in \mathbb{N}$. Using (16), we have

$$\begin{aligned} g_{m+1}(\omega) &= \int_0^{\omega} g_m(t)\,dt \\ &\overset{(9)}{=} \int_0^{\omega} \left\{ (t-1)^{m-1} \left(z_{m-1} - \frac{\ln(1-t)}{(m-1)!} \right) - \sum_{j=0}^{m-2} \frac{(-1)^{m-1-j}}{j!} z_{m-1-j} t^j \right\} dt. \end{aligned}$$

Integration by parts yields

$$\begin{aligned} g_{m+1}(\omega) = \int_0^{\omega} g_m(t)\,dt &= \frac{(t-1)^m}{m} \left(z_{m-1} - \frac{\ln(1-t)}{(m-1)!} \right) \Bigg|_0^{\omega} \\ &+ \int_0^{\omega} \frac{(t-1)^{m-1}}{m!}\,dt - \sum_{j=0}^{m-2} \frac{(-1)^{m-1-j}}{(j+1)!} z_{m-1-j} \omega^{j+1} \end{aligned}$$

$$
\begin{aligned}
&= (\omega-1)^m \left(\frac{z_{m-1}}{m} - \frac{\ln(1-\omega)}{m!} \right) - (-1)^m \frac{z_{m-1}}{m} \\
&\quad + \frac{(\omega-1)^m}{m!\,m} - \frac{(-1)^m}{m!\,m} - \sum_{j=1}^{m-1} \frac{(-1)^{m-j}}{j!} z_{m-j}\omega^j \\
&= (\omega-1)^m \left(\frac{z_{m-1}}{m} + \frac{1}{m!\,m} - \frac{\ln(1-\omega)}{m!} \right) \\
&\quad - (-1)^m \frac{z_{m-1}}{m} - \frac{(-1)^m}{m!\,m} - \sum_{j=1}^{m-1} \frac{(-1)^{m-j}}{j!} z_{m-j}\omega^j \\
&\overset{(10)}{=} (\omega-1)^m \left(z_m - \frac{\ln(1-\omega)}{m!} \right) - (-1)^m z_m - \sum_{j=1}^{m-1} \frac{(-1)^{m-j}}{j!} z_{m-j}\omega^j \\
&= (\omega-1)^m \left(z_m - \frac{\ln(1-\omega)}{m!} \right) - \sum_{j=0}^{m-1} \frac{(-1)^{m-j}}{j!} z_{m-j}\omega^j .
\end{aligned}
$$

Acknowledgments

The author is grateful to Ingo Eble for the programming of the numerical software. He would also like to thank the referees for helpful comments.

REFERENCES

Alefeld, G. and Herzberger, J. 1983. *Introduction to interval computations.* Academic Press, New York.

Bühler, G. 1993. *Standardfunktionen für komplexe Intervalle im 64 Bit IEEE Datenformat.* Diploma thesis, Universität Karlsruhe.

Braune, K. and Krämer, W. 1987. High-accuracy standard functions for real and complex intervals. In E. Kaucher, U. Kulisch, and C. Ullrich Eds., *Computerarithmetic: Scientific computation and programming languages* (1987), pp. 81–114. Teubner, Stuttgart.

Eble, I. and Neher, M. 2001. *Computation of validated bounds for Taylor coefficients of analytic functions with ACETAF.* Preprint 01/18, Institut für Wissenschaftliches Rechnen und Mathematische Modellbildung, Universität Karlsruhe.

Geddes, K. O. and Mason, J. C. 1975. Polynomial approximation by projections on the unit circle. *SIAM J. Numer. Anal. 12*, 111–120.

Griewank, A. 2000. *Evaluating derivatives: Principles and techniques of algorithmic differentiation.* SIAM, Philadelphia.

Hansen, E. R. 1992. *Global optimization using interval analysis.* Dekker, New York.

Henrici, P. 1974. *Applied and computational complex analysis, Vol. I.* Wiley, New York.

Jaulin, L., Kieffer, M., Didrit, O., and Walter, E. 2001. *Applied interval analysis.* Springer, London.

Kearfott, R. B. 1996. *Rigorous global search: Continuous problems.* Kluwer, Dordrecht.

Klatte, R., Kulisch, U., Lawo, C., Rauch, M., and Wiethoff, A. 1993. *C-XSC: A C++ class library for extended scientific computing.* Springer, Berlin.

Kulisch, U. and Miranker, W. L. 1981. *Computer arithmetic in theory and practice.* Academic Press, New York.

Lerch, M., Tischler, G., and Wolff von Gudenberg, J. 2001a. filib++ - Interval library specification and reference manual. Technical Report 279, Universität Würzburg.

Lerch, M., Tischler, G., Wolff von Gudenberg, J., Hofschuster, W., and Krämer, W. 2001b. The interval library filib++ 2.0. Design, features and sample programs. Preprint 2001/4, Universität Wuppertal, Wissenschaftliches Rechnen/Softwaretechnologie.

Moore, R. E. 1966. *Interval analysis.* Prentice Hall, Englewood Cliffs, N.J.

Neher, M. 1997. Enclosing power series solutions of ODEs. *ZAMM 77*, S2, S635–S636.

Neher, M. 2001a. Berechenbare Schranken von Taylorkoeffizienten analytischer Funktionen. *ZAMM 81*, S3, S747–S748.

Neher, M. 2001b. Geometric series bounds for the local errors of Taylor methods for linear n-th order ODEs. In G. Alefeld, J. Rohn, S. Rump, and T. Yamamoto Eds., *Symbolic Algebraic Methods and Verification Methods* (2001), pp. 183–193. Springer, Wien.

Neher, M. 2001c. Validated bounds for Taylor coefficients of analytic functions. *Reliable Computing 7*, 307–319.

Neumaier, A. 1990. *Interval methods for systems of equations.* Cambridge University Press, Cambridge.

Rall, L. B. 1981. *Automatic differentiation: Techniques and applications, Lecture Notes in Computer Science, Vol. 120.* Springer, Berlin.

Ratschek, H. and Rokne, J. 1988. *New computer methods for global optimization.* Ellis Horwood Limited, Chichester.

Sakurai, T. and Sugiura, H. 2000. On factorization of analytic functions and its verification. *Reliable Computing 6*, 459–470.

Walsh, J. L. 1935. *Approximation by polynomials in the complex domain.* Gauthier-Villars, Paris.