Journal of Mathematical Research and Exposition
Vol.11 No.1 Feb. 1991

Roundoff Error Analysis of Algorithms for Polynomial Interpolation*

Huamg Kaibing *Li Zhilin*

(Dept. Math., Nanjing Normal University)

Abstract

The condition of a polynomial interpolation operator and the numerical stability of an algorithm for polynomial interpolation are defined. The main result is that both Newton and Lagrange interpolation algorithms are numerically stable provided that the nodes are evenly distributed, but the extrapolation computation of these methods will lose the accuracy whatever the nodes are.

1 Basic Concepts

It is necessary to analyse the roundoff errors and the credibility of the computed results for polynomial interpolation. For example, given $y=f(x)=(1.01)^x$, $x\in[0,1.5]$ and 11 support points (x_i, y_i), $i=0,1,\cdots,10$ where $x_i=0.01\times i$, $i=0,1,\cdots 9$, $x_{10}=1.5$, $y_i=(1.01)^{x_i}$, $i=0,1,\cdots,10$. We denote the Lagrange or Newton polynomial whose degree dose not exceed 10 by $p_{10}(x)$ for which $p_{10}(x_i)=f(x_i)$, $i=0,1,\cdots,10$. If only the truncation error $|p_{10}(x)-f(x)|$, $x\in[0,0.2]$ bounded by $7E-14$, is considered, $p_{10}(x)$ is a good approximation of $f(x)$, $x\in[0,0.2]$. But if we calculate $p_{10}(x)$ on the computer with 7 decimal digits, the computed results contain only one significant digit when $x\in[0.136,0.16]$ and they have no significant digit at all when $x\in[0.164,0.2]$.

Now we present some basic concepts. For a given set of support points

$$(X,Y)=\{(x_i,y_i)\,|\,x_i\neq x_j,\ \text{for}\ i\neq j, i=0,1,\cdots,n\} \tag{1}$$

and the set of polynomials $q_n(x)$ of degree $\leq n$

$$Q_n=\{q_n(x)\,|\deg(q_n(x))\leq n\}. \tag{2}$$

Definition 1 The polynomial interpolation operator $P(X,Y)$

$$P(X,Y):\ (X,Y)\to p_n(x)\in Q_n,\ X\in I[x_0,x_1,\cdots,x_n],\ p_n(x_i)=y_i,\ i=0,1,\cdots,n \tag{3}$$

is called *ill-conditioned* if the solution $p_n(x)$ to be computed is very sensitive to small changes in the data (X,Y). otherwise $P(X,Y)$ is called *well-conditioned*.

* Received Jan. 16, 1989. The project supported by National Natural Science Foundation of China.

Consider the corresponding perturbed problem of (1)—(3)

$$(X+\Delta X, Y+\Delta Y)=\{(x_i+\varepsilon_i, y_i+\eta_i)\,|\,x_i+\varepsilon_i\neq x_j+\varepsilon_j, i\neq j, i=0,1,\cdots,n\}$$

$$P(X+\Delta X, Y+\Delta Y):\ (X+\Delta X, Y+\Delta Y)\to p_n(x)+\delta p_n(x).$$

And $p_n(x)$ in (3) can be expressed by the basis polynomials $\{\psi_j(x)\}_0^n$ of Q_n:

$$p_n(x)=\sum_{j=0}^{n} y_j\psi_j(x). \tag{4}$$

From the Taylor theorem of multivariable function, we have

$$|\delta p_n|\leq\varepsilon\sum_{i=0}^{n}(|\psi_i(x)|+|\sum_{j=0}^{n} y_j\partial\psi_j(x)/\partial x_j|)+O(\varepsilon^2)\triangleq\varepsilon C_p(x, X, Y)+O(\varepsilon^2) \tag{5}$$

where $\varepsilon=\max(|\varepsilon_i|, |\eta_i|, i=0,1,\cdots,n)$. $C_p(x, X, Y)$ can describe the sensitivity of $P(X, Y)$ in (3), and is called the *condition number* of polynomial interpolation operator $P(X, Y)$.

In floating computation we use the set of floating representations of (1)

$$(fl(X), fl(Y))=\{(fl(x_i), fl(y_i)),\ i=0,1,\cdots,n\}, \tag{1'}$$

instead of (1). We suppose that $fl(x_i)\neq fl(y_i)$ for $i\neq j$. Let $p_n^*(x)$ and $\overline{p_n}(x)$ obtained by an algorithm denote the exact and computed polynomial of the interpolation problem (1'), (2) and (3) respectively. Since $p_n(x)$ is the exact solution of problem (1)—(3), hence

$$|p_n(x)-\overline{p_n}(x)|\leq|p_n(x)-p_n^*(x)|+|p_n^*(x)-\overline{p_n}(x)|. \tag{6}$$

obviously $p_n(x)-p_n^*(x)$ results from the data input errors, but $p_n^*(x)-\overline{p_n}(x)$ comes from the roudoff errors produced by carrying out algorithim (A). Let ζ denote the machine precision, and $\varepsilon=C\zeta$, where C is a constant. Ignoring the high order terms of ζ, we have approximately

$$|p_n(x)-\overline{p_n}(x)|\leq\zeta CC_p(x, X, Y)+\zeta C_R(x, X, Y). \tag{7}$$

Definition 2 Algorithm (A) is numerically stable if $p_n^*(x)-\overline{p_n}(x)$ has the same order of magnitude as $\zeta C_R(x, X, Y)$, where $C_R(x, X, Y)$ is a bounded function.

Notice that if $C_p(x, X, Y)$, the inherent property of a given interpolation problem is so large that $\overline{p_n}(x)$ can not attain the given accuracy, it is impossible to improve the accuracy of computed solution only by studying the algorithm.

2 Main Results

From now on suppose that $x_i's, y_i's$ of (1) are standard floating point numbers or zeros. We will consider Algorithm 1 for carrying out the Lagrange interpolation formula:

$$p_n(x)=\sum_{i=0}^{n}\psi_i(x)\,y_i,\ \psi_i(x)=\prod_{\substack{j=0\\ j\neq i}}^{n}(x-x_j)/(x_i-x_j),\quad x\in I[x_0,\cdots,x_n].$$

Algorithm 1 Set $p_n(x)\Leftarrow 0$.

For $i=0,1,\cdots,n$, compute

$\psi_i=1$;

For $j=0,1,\cdots,n$, $j\neq i$, compute

$$\psi_i=\psi_i(x-x_j)/(x_i-x_j),$$

$p_n(x)\leftarrow p_n(x)+\psi_i y_i$.

Theorem 1 For the interpolation problem (1)—(3), let x be a standard floating point number or zero and $\overline{p_n}(x)$ be the computed result by Algorithm 1. Then

$$|p_n(x)-\overline{p_n}(x)|\leq\zeta C_L(x,X,Y)(5n+1)+O(\zeta^2), \tag{8}$$

where

$$C_L(x,X,Y)=\sum_{i=0}^{n}|\psi_i(x)y_i|. \tag{9}$$

Proof From formulae of roundoff errors of the fundamental floating poing arithmatic operations, we have

$$fl(\psi_i(x))=\psi_i(x)(1+E_i^{(1)}),\quad \overline{p_n}(x)=\sum_{i=0}^{n}\psi_i(x)y_i(1+E_i^{(2)}). \tag{10}$$

where

$$(1-\zeta)^{4n-1}\leq 1+E_i^{(1)}\leq(1+\zeta)^{4n-1},\quad (1-\zeta)^{5n-i+2}\leq 1+E_i^{(2)}\leq(1+\zeta)^{5n-i+2} \tag{11}$$

It follows

$$|p_n(x)-\overline{p_n}(x)|\leq\zeta\Big|\sum_{i=0}^{n}\psi_i(x)y_iE_i^{(2)}\Big|\leq\zeta(5n+1)\sum_{i=0}^{n}|\psi_i(x)y_i|+O(\zeta^2). \quad \blacksquare$$

Here the quantity $C_L(x,X,Y)=\sum_{i=0}^{n}|\psi_i(x)y_i|$ truly reflects the numerical stability of Algorithm 1.

Corollary 1 Let $x_i=x_0+ih$, $i=0,1,\cdots,n$, where $h>0$ and $x\in[x_0,x_n]$ are either standard floating point numbers or zeros. Then

$$|p_n(x)-\overline{p_n}(x)|\leq(5n+1)\zeta Y_{\max}2^n+O(\zeta^2),\ Y_{\max}=\max_i\{|y_i|\}. \tag{12}$$

Proof Let $\Pi_i(x)=\prod_{\substack{j=0\\ j\neq i}}^{n}(x-x_j)$, we can easily prove that $|\Pi_i(x)|\leq n!h^n$, $x\in[x_0,x_n]$. Therefore we can immediately obtain that $|\psi_i(x)|\leq C_n^i$, $i=0,1,\cdots,n$. Finally it follows

$$|p_n(x)-\overline{p_n}(x)|\leq(5n+1)\zeta Y_{\max}\sum_{i=0}^{n}C_n^i+O(\zeta^2)\leq(5n+1)\zeta Y_{\max}2^n+O(\zeta^2). \quad \blacksquare$$

The right hand side of (12) contains the factor 2^n, it explains the unstability of high order interpolation. So generally the degree of interpolation is restricted (say $n\leq 6$). Under such condition, the Lagrange algorithm for the interpolation problem with equally spaced nodes is numerically stable.

It is well known that the Newton interpolation formula depends on the divided difference scheme, generally speaking, in floating point computation the

symmetry of the divided difference is no longer true.

Algorithm 2

For $k=0,1,\cdots,n$, set $y(x_k)=y_k$,

For $i=1,2,\cdots,n$, $k=0,1,\cdots,n-i$, compute

$$y(x_k,\cdots,x_{k+i})=(y(x_k,\cdots,x_{k+i-1})-y(x_{k+1},\cdots,x_{k+i}))/(x_k-x_{k+i}). \tag{13}$$

The computations take place in order of (13).

Theorem 2 Let $\overline{y}(x_k,\cdots,x_{k+i})$ be the computed result by carrying out Algorithm 2. Then

$$|(\overline{y}(x_k,\cdots,x_{k+i})-y(x_k,\cdots,x_{k+i})|\leq\zeta\sum_{j=0}^{i}y_{j+i}|y_{k+j}/\prod_{m\neq j}^{i}(x_{k+j}-x_{k+m})|, \tag{14}$$

where

$$y_{j,i}=\begin{cases}3, & i=1 \text{ and } j=0,1\\ y_{j,i-1}+3, & j=0\\ y_{j,i-1}+y_{j-1,i-1}+3, & j=1,2,\cdots,i-1\\ y_{i-1,i-1}+3, & j=1.\end{cases} \tag{15}$$

Furthermore, we have

$$|y_{j,i}|<(2.01)^{i+2}. \tag{16}$$

Proof The proof can be completed by induction and from the fact that $2(2.01)^p+3\leq(2.01)^p$ for $p>9$. Here we omit the detail. ■

Corollary 2 Let $M_i=\max\{|1/\prod_{m\neq j}^{i}(x_j-x_m)|\}$, $Y_{\max}=\max_{0\leq j\leq i}\{|y_j|\}$. Then

$$|\overline{y}(x_0,\cdots,x_i)-y(x_0,\cdots,x_i)|\leq\zeta(2.01)^{i+2}(i+1)M_iY_{\max}. \tag{7}$$

The proof is trivial.

We can easily prove the following theorem loy induction:

Theorem 3 Suppose that $x_i=x_0+ih$, $i=0,1,\cdots,n$, $h>0$, then the computed result $\overline{y}(x_k,\cdots,x_{k+i})$ by Algorithm 2 satisfies:

$$|\overline{y}(x_k,x_{k+1},\cdots,x_{k+i})-y(x_k,x_{k+1},\cdots,x_{k+i})|\leq\zeta Y_{\max}2^{i+2}i/(i!h^i). \tag{18}$$

Theorem 3 shows that Algorithm 2 is numerically stable, so the result of Theorem 3 is quite satisfactory.

Algorithm 3

Compute $y(x_0,x_1,\cdots,x_i)$, $i=0,1,\cdots,n$ by Algorithm 2,

$p_n(x)\Leftarrow y_0$,

For $i=1,2,\cdots,n$

$T_i\Leftarrow y(x_0,x_1,\cdots,x_i)$,

For $j=1,\cdots,i-1$,

$T_i=T_i(x-x_j)$

$p_n(x)\Leftarrow p_n(x)+T_i$.

From Theorem 2 and through simple roundoff error analysis, we have:

Theorem 4 For interpolation problems (1)—(3), let $x \in I[x_0, \cdots, x_n]$ be a standard floating point number or zero, and $\overline{p_n}(x)$ be the computed result by Algorithm 3. Then

$$|p_n(x) - \overline{p_n}(x)| \leq \zeta C_N(x, X, Y)(2.01^{n+2} + 2n + 1) + \overline{O(\zeta^2)}, \tag{19}$$

where

$$C_N(x, X, Y) = \sum_{i=0}^{n} \sum_{j=0}^{i} |y_j(\prod_{k=0}^{i-1}(x - x_k)/\prod_{k\neq j}^{i}(x_j - x_k))|. \tag{20}$$

Quantity $C_N(x, X, Y)$ reflects the numerical stability of Algorithm 3 for Newton interpolation. But the most leading factor for determining the numerical stability of Algorithm 3 is $C_N(x, X) = \max\limits_{\substack{0\leq i\leq n-1 \\ 0\leq j\leq i}} |\prod_{k=0}^{i-1}(x - x_k)/\prod_{k\neq j}^{i}(x_j - x_k)|$.

The following theorem shows that the Newton algorithm with equally spaced nodes is numerically stable whenever n is not too large.

Theorem 5 Suppose that $x_i = x_0 + ih$, $h > 0$, $i = 0, 1, \cdots, n$. Let $x \in [x_0, x_n]$ be the standard floating point number or zero, and $\overline{p_n}(x)$ be the computed solution by Algorithm 3 for interpolation problem (1)—(3). Then

$$|p_n(x) - \overline{p_n}(x)| \leq Y_{max}\zeta(6n + 1)3^n. \tag{21}$$

Proof Since

$$|y(x_0, x_1, \cdots, x_i)| \leq Y_{max}2^i/(i!h_i), \tag{22}$$

and

$$|\prod_{k=0}^{i-1}(x - x_k)|/(i!h^i) \leq C_n^i, \quad i = 0, 1, \cdots, n. \tag{23}$$

(21) follows from Theorem 3 and 4. ■

As mentioned above there are two conclusions: 1. If n and Y_{max} are not too large, both Lagrange and Newton interpolation algorithms with equally spaced nodes are numerically stable. 2. The extrapolation computation of these methods may be numerically unstable whatever the nodesare. It is clear that $C_L(x, X, Y)$ and $C_N(x, X, Y)$ increase rapidly when x is out of the interpolation interval.

3 Numerical Examples

We have tested some numerical examples on an IBM-PC computer. Below let $p_L(x)$ ($p_N(x)$), $\overline{p_L}(x)$ ($\overline{p_N}(x)$) denote the exact and computed results of Lagrange (Newton) interpolation algorithm respectively. Those figures underlined indicate that the accuracy by Lagrange algorithm is higher than Newton's.

Ex 1 Given for $n = 10$, $x_i = 0.01 \times i$, $i = 0, 1, \cdots, 9, x_{10} = 1.5, y = e^{(1+x_i)}$, $i = 0, 1, \cdots, 10$. Wanted: $p_{10}(x)$ for which $p_{10}(x_i) = y_i$, $i = 0, 1, \cdots, 10$.

Table 1

x	$\lvert\overline{p}_L(x)-p_n(x)\rvert$	$\lvert\overline{p}_N(x)-p_n(x)\rvert$	$C_L(x, X)$
.005	1.204 $E-6$	2.04 $E-7$	4.16 $E1$
.055	5.3 $E-8$	5.3 $E-8$	9.0
.085	7.26 $E-7$	2.74 $E-7$	3.97 $E1$
.093	2.34 $E-5$	1.65 $E-6$	2.3 $E2$
.095	2.3 $E-5$	4.3 $E-6$	5.8 $E2$
.1	4.36 $E-6$	2.36 $E-5$	2.96 $E3$
.11	7.49 $E-4$	2.1 $E-4$	2.7 $E4$
.12	4.52 $E-3$	1.05 $E-3$	1.38 $E6$
.15	7.72 $E-2$	3.14 $E-2$	4.29 $E7$
.2	1.763	1.2	1.7 $E8$
.25	15.7	15.1	2.16 $E9$
.1	7.63 $E7$	6.5 $E6$	9.64 $E14$

Ex 2 Given $x_i = i$, $y_i = 100 \times \sin(x_i)$, $i = 0, 1, \cdots, 9$, and $x_{10} = 100$, $y_{10} = 200$. Wanted: $p_{10}(x)$.

Table 2

x	$\lvert\overline{p}_L(x)-p_n(x)\rvert$	$\lvert\overline{p}_N(x)-p_n(x)\rvert$	$C_L(x, X)$
.5	5.46 $E-4$	2.46 $E-4$	3.15 $E3$
.8	4.62 $E-4$	6.24 $E-5$	1.2 $E3$
1.1	1.67 $E-4$	3.29 $E-5$	7.55
2.5	1.23 $E-4$	2.25 $E-5$	8.38 $E2$
8.5	2.87 $E-4$	4.49 $E-3$	2.92 $E3$
9.2	5.02 $E-4$	7.6 $E-3$	8.93 $E3$
9.5	2.58 $E-4$	9.42 $E-4$	4.3 $E4$
10	1.2 $E-3$	1.83 $E-3$	2.17 $E5$
11	0.21	0.12	1.98 $E6$
12	1.24	0.57	1 $E7$
14	34.84	6.86	1.15 $E8$
101	4.4 $E8$	1.39 $E8$	2.4 $E15$

Ex 3 Determine $p_{10}(x)$ satisfying $y_i = f(x_i) = (1.01)^{x_i}$, $x_i = 0.01 \times i$, $i = 0, \cdots, 9$, $x_{10} = 1.5$.

Table 3

x	or $\frac{\lvert\overline{p}_L(x)-p_n(x)\rvert}{\lvert\overline{p}_L(x)-f(x)\rvert}$	or $\frac{\lvert\overline{p}_N(x)-p_n(x)\rvert}{\lvert\overline{p}_N(x)-f(x)\rvert}$	$C_L(x, X)$	$\lvert p_n(x)-f(x)\rvert$
$-.085$	$8.1E-2$	0.248	$3.2E6$	$3.1E-14$
$-.03$	$1.5E-3$	$9.6E-4$	$1.24E4$	$1.2E-16$
$-.02$	$5.8E-4$	$1.87E-4$	$2.4E3$	$2.31E-17$
$-.01$	$5.3E-5$	$2.1E-5$	$2.6E2$	$2.5E-18$
$.0001$	$4.7E-7$	0	2.5	$2.4E-20$
$.065$	$2.4E-7$	0	$9E-1$	$8.7E-21$
$.085$	0	$2.4E-7$	3.3	$3.2E-20$
$.1$	$9.4E-6$	$1.75E-5$	$2.4E2$	$2.3E-18$
$.12$	$6.1E-4$	$8.4E-4$	$1.13E4$	$1.08E-16$
$.138$	$1.5E-2$	$7.97E-3$	$1.1E5$	$1.05E-15$
$.15$	$6.7E-2$	$2.65E-2$	$3.5E5$	$3.3E-15$
$.166$	$3.7E-1$	$1E-1$	$1.4E6$	$1.33E-14$
$.187$	2.6	$4.5E-1$	$6.2E6$	$5.6E-14$
$.2<x<1.49$	Failure		$[E6, 6E13]$	$[6E-14, 6E-7]$

References

[1] J. H. Wilkinson, Rounding Errors in Algebraic Processes, Prentice-Hall, Inc., 1963.

[2] ——, The Algebraic Eigenvalue Problem, Oxford University Press, 1965.

[3] J. Stoer and R. Bulirsch, Introduction to Numerical Analysis, Springer-Verlag New York Inc., 1980.

[4] Youqian Huang and Yuesheng Li, Numerical Approximation, Advanced Educetion Press, 1987.

[5] Yuanji Zhang and Kaibing Huang, Modern Ronuding Error Analysis, Nanjing University Press, 1990.

多项式插值算法的舍入误差分析

黄开斌　　李治林

摘　要

本文定义了多项式插值算子的条件数和多项式插值算法的数值稳定性等概念．主要研究结果是：若n和Y_{max}不太大，当结点等距分布时，Lagrange插值和Newton插值算法都是数值稳定的．但是不论结点如何分布，上述两法的外推计算可能是数值不稳定的．文中数值例子验证了这些理论结果．