On the Role of Computable Error Estimates in the Analysis of Numerical Approximation Algorithms -- In Honor of Steve Smale on His Sixtieth Birthday

by

Feng Gao

Technical Report 91-7
June, 1991

Department of Computer Science
University of British Columbia
Vancouver, B.C.
CANADA V6T 1Z2

On the Role of Computable Error Estimates in the Analysis of Numerical Approximation Algorithms -- In Honor of Steve Smale on His Sixtieth Birthday

Feng Gao

Department of Computer Science
University of British Columbia
Vancouver, B. C. V6T 1W5, Canada

and
Nankai Institute of Mathematics
Tianjin, China

Truncation error estimation for methods of numerical approximation, i.e., estimation of their errors of approximation, is an important constituent of numerical analysis. The error estimates obtained can be dichotomized according to whether an error estimate is computable from information that can be used for computing the approximations. For example, the standard error estimate for the Bisection method for finding a zero of a function $f(x)$ in an interval $[a,b]$, under the assumption that f is continuous on $[a,b]$ with $f(a)\,f(b) < 0$, is computable. The method proceeds as follows. Let $[a_0,b_0] = [a,b]$. At Step k ($k \geq 0$), compute $f(x_k)$, where $x_k = \frac{a_k+b_k}{2}$. If $f(x_k) = 0$, an exact zero is found. Otherwise let $[a_{k+1},b_{k+1}] = [a_k,x_k]$ if $f(x_k) > 0$, or $[a_k,b_k] = [x_k,b_{k-1}]$ if $f(x_k) < 0$, and proceed to Step $k+1$. Now let x_k be the kth approximation to an zero of f in $[a,b]$. Then there exists $\bar{x} \in [a,b]$ such that $f(\bar{x}) = 0$, and

$$|\bar{x} - x_k| \leq \frac{(b-a)}{2^k}. \tag{1}$$

The Right-Hand-Side of (1) is a computable and guaranteed (deterministic) bound on the error of approximation.

An example of a non-computable error estimate is the standard one for the trapezoidal rule for approximating the integral of a function $f(x)$ over $[a,b]$,

$$\int_a^b f(x)\,dx \;\approx\; \frac{(b-a)}{2}\{f(a) + f(b)\}.$$

For any $f \in C^2[a,b]$,

$$\left|\int_a^b f(x)\,dx - \frac{(b-a)}{2}\{f(a)+f(b)\}\right| \;\leq\; \frac{(b-a)^3}{12} \sup_{a\leq x\leq b} |f^{(2)}(x)|. \tag{2}$$

The term $\sup_{a\leq x\leq b} |f^{(2)}(x)|$ in the Right-Hand-Side of (2) is not computable for an arbitrary

C^2-function, i.e., it cannot be computed using a finite amount of data from the set $\{x, f(x)\}_{x\in[a,b]}$ as does a quadrature approximation.

This simple dichotomy of truncation error estimates according to whether an estimate is computable had not seemed to attract much attention in traditional numerical analysis. Its potential importance was perceived by Steve Smale and became the basis of his 1986 paper "Newton's method estimates from data at one point" (Smale (1986)). Smale commented at the beginning of this paper:

> The work of Kantorovich has been seminal in extending and codifying Newton's method. Kantorovich's approach, which dominates the literature in this area, has these features: (a) weak differentiability hypotheses are made on the system, e.g., the map is C^2 on some domain in a Banach space; (b) derivative bounds are supposed to exist over the whole of this domain. In contrast, here strong hypotheses on differentiability are made; analyticity is assumed. On the other hand, we deduce consequences from data at a single point. This point of view has valuable features for computation and its theory.

Smale then presented a criterion for testing whether an approximation z_0 to a zero of a complex polynomial or complex analytic function f is a so-called approximate zero. More precisely, the criterion involves estimation of the quantity $\alpha(z,f)$,

$$\alpha(z,f) = \left|\frac{f(z)}{f'(z)}\right| \sup_{k>1} \left|\frac{f^{(k)}(z)}{k!f'(z)}\right|^{1/k-1} \tag{3}$$

at $z = z_0$ to see whether it is less than $\alpha_0 \approx 0.130707$. If the criterion is satisfied, then z_0 is an approximate zero, which is defined as satisfying the condition that Newton's iteration $z_k = z_{k-1} - f(z_{k-1})/f'(z_{k-1})$ starting at z_0 has the property

$$|z_k - z_{k-1}| \leq (\frac{1}{2})^{2^{k-1}-1}|z_1 - z_0|, \qquad k = 1, 2, \cdots \tag{4}$$

It was then shown that if z_0 is an approximate zero then there exist η, $f(\eta) = 0$, such that

$$|z_n - \eta| \leq (\frac{1}{2})^{2^n-1}|z_1 - z_0|\, K, \qquad n = 1, 2, \cdots \tag{5}$$

where $K \leq 1\frac{3}{4}$. In other words, starting at an approximate zero, the Newton iteration approximation is super-convergent. Here the criterion (3) is computable from the coefficients of the analytic series of f at z_0 (when f is not a polynomial, I use the term "computable" just to mean dependence on data only at the single point z_0, since the supremum is taken over a countable set of numbers), and estimates (4) and (5) do not involve any non-computable, function-dependent constant. This is in contrast to the traditional Kantorovich theory (Kantorovich and Akilov (1964)), where, similar to the case of the trapezoidal rule for integral approximation, the convergence estimate involves a bound on the second derivative of f over a whole domain which is not computable locally for an arbitrary C^2-function. The new, computable estimate was used by Smale to construct new zero-finding algorithms based on Newton's method.

Even though this so-called α-theory for zero-finding of analytic functions can also be derived from the Kantorovich theory (e.g., Rheinbolt (1988)), The significance of Smale's observation on computable error estimates goes far beyond zero-finding. This is because, while a non-computable error estimate provides information regarding the speed of convergence of a method of approximation, a computable error estimate can also be used as an error criterion in an algorithm. This is a very important point which I shall return to later in this paper.

In Fall 1985 at Berkeley, when Steve Smale was disseminating this α-theory in his graduate course and also in one Mathematics Department colloquium talk, I was at the preliminary stage of my Ph. D. research under his supervision. Smale's observation on computable error estimates struck me as pointing out a potentially important direction for the analysis of numerical approximation algorithms. More thinking brought me to the following conclusion: most numerical approximation methods simply do no have guaranteed and yet computable error bounds under a weak differentiability assumption; while a strong differentiability assumption is one way to obtain computable error estimates, estimates applicable under a weak differentiability assumption is also important, for this assumption usually captures the generality of a method and is the starting point of many algorithms in practice; most practical algorithms thus use computable but not guaranteed error estimates; they are heuristics that may be incorrect some of the time but prove to be generally useful in practice.

Around this time, I was studying Smale's 1985 paper "On the efficiency of algorithms of analysis" (Smale (1985)) and in particular the section "On efficient approximation of integrals". I was also reading a preprint of the paper "Approximation of linear functionals on a Banach space with a Gaussian measure" by David Lee and Greg Wasilkowski (Lee and Wasilkowski (1986)). Both concern the average analysis of integral approximation using Gaussian measures on function spaces. The former focused on the average errors of some classical quadratures while the latter focused more on the average approximative power of quadratures for a given number of allocation points. Upon Smale's suggestion, I was also exploring the problem of adaption for numerical integration in this average analysis setting, which resulted in an unpublished manuscript Gao (1986). It then occurred to me that my thoughts motivated by Smale's observation on computable error estimates could perhaps be formalized in this average analysis setting. Namely, the assumption of a probability measure on a function space with weak differentiability may lead to computable conditional average error estimates -- bounds on the average error given the computed data -- for a approximation method, even though computable and guaranteed error bounds may not exist. And this formalism may serve as the theoretical basis for the heuristic error estimates in practical algorithms.

I decided that the key was to first show the relevance of such a formalism by establishing a close connection between the conditional average error estimates produced in such a mathematical model and the practical heuristic estimates. The case of the trapezoidal rule mentioned earlier was the right canonical example for me since I was looking at numerical integration. More specifically, I considered the following problem:

The integral $\int_a^b f(t)\, dt$ is to be evaluated for $f \in C_0^k[a,b]$, where

$$C_0^k[a,b] = \{ g \in C^k[a,b] : g^{(i)}(a) = 0, \ i = 0, 1, \ldots, k \}.$$

A quadrature $Q_n(f) = \sum_{i=1}^n a_i f(t_i)$ is used to approximate the integral, where $a = t_0 < t_1 < \cdots < t_n = b$ and the choices of the quadrature points $\{ t_i \}_{i=1}^n$ and the coefficients $\{ a_i \}_{i=1}^n$ depend on n and satisfy the following two conditions:

Condition 1 There exists a constant $c > 0$, independent of n, such that

$$\frac{\max_{1 \le i \le n} | t_i - t_{i-1} |}{\min_{1 \le i \le n} | t_i - t_{i-1} |} \le c \tag{6}$$

Condition 2 There exists a constant $\bar{c} > 0$, independent of n, such that

$$\left| \int_a^b f(t)dt - \sum_{i=1}^n a_i f(t_i) \right| \le \frac{\bar{c}}{n^k} \sup_{a \le t \le b} | f^{(k)}(t) | \tag{7}$$

for any $f \in C_0^k[a,b]$.

Since $\sup_{a \le t \le b} | f(t) |$ is not finitely computable from $\{x, f(x)\}_{x \in [a,b]}$, we suppose an algorithm employs the following heuristic error criterion obtained by replacing the kth derivative with the kth divided differences:

$$\left| \int_a^b f(t)dt - \sum_{i=1}^n a_i f(t_i) \right| \lesssim \frac{\bar{c}}{n^k} \max_{1 \le i \le n} | [t_{i-k}, \ldots, t_i] f | \tag{8}$$

Here $[t_{i-k}, \ldots, t_i] f$ denotes the k-th divided difference of f at $t_{i-k}, \cdots, t_i$ and by convention $t_{-i} = a - t_i$, $f(t_{-i}) = 0$, $i = 1, \ldots, n$. Also assume $n > k$. The (recursive) definition of the divided difference is given by

$$[t_{i-l}, \ldots, t_i] f = \frac{[t_{i-l+1}, \ldots, t_i] f - [t_{i-l}, \ldots, t_{i-1}] f}{t_i - t_{i-l}}$$

$$i = 1, \ldots, n, \quad 1 \leq l \leq n.$$

The criterion (8) is not a guaranteed error bound since the actual error of approximation can be arbitrary large for a C^k-function whose kth divided differences at $\{t_i\}$ are small. It is a variant of some of the heuristic error criteria used in practical quadrature algorithms.

To analyze (8), I then assumed that the likelihood of functions in $C_0^k[a,b]$ as integrands are distributed according to the Wiener measure in $C_0^k[a,b]$. Intuitively, this measure is a Gaussian measure on $C_0^k[a,b]$ with mean $f \equiv 0$ and variance 1, and can be viewed as concentrating in

$$H_0^{k+1}[a,b] = \{ f \in H^{k+1}[a,b]\colon \ f^{(i)}(a) = 0, \ i = 0, 1, \ldots, k \}$$

with the normal probability density distributed with respect to the norm

$$\|f\|_{H_0^{k+1}} = \left(\int_a^b (f^{(k+1)}(t))^2 dt \right)^{\frac{1}{2}}$$

where

$$H_0^{k+1}[a,b] \ = \ \{ f \in C_0^k[a,b] : \ f^{(k+1)} \in L_2[a,b] \}$$

is a Hilbert space with inner product $<f,g> = \int_a^b f^{(k+1)}(t) g^{(k+1)}(t) \, dt.$

The following theorem became the main result of my Ph. D. thesis.

Theorem (Gao (1989)) There exist constants $c_1 \geq 0$, and $c_2 > 0$, such that for any quadrature $\sum_{i=1}^{n} a_i f(t_i)$ which satisfies (6) and (7),

$$\underset{\substack{g \in C_0^k[a,b] \\ g(t_i)=f(t_i) \\ i=1,\ldots,n}}{Average} \left| \int_a^b g(t)\,dt - \sum_{i=1}^{n} a_i g(t_i) \right|^2 \leq \frac{c_1}{n^{2k+2}} + c_2 \, n \left(\frac{\bar{c}}{n^k} \max_{1 \leq i \leq n} | [\, t_{i-k}, \ldots, t_i \,] f | \right)^2 \tag{9}$$

for any $(f(t_1), \ldots, f(t_n))^T \in R^n$. Here $\underset{\substack{g \in C_0^k[a,b] \\ g(t_i)=f(t_i) \\ i=1,\ldots,n}}{Average}$ denotes the average with respect to the conditional Wiener measure under the constraints $g(t_i) = f(t_i)$, $i = 1, 2, \cdots, n$.

Condition 1 on a uniform partitioning of $[a,b]$ is too stringent for practice. In a later paper Gao (1990b), it was relaxed and Condition 2 was replaced with a sum of local error bounds, allowing the theory to cover adaptive quadratures.

This theorem demonstrated the close connection between the probabilistic model and the heuristic error estimates in practice. The technical results needed to prove it also gave rise to new numerical quadrature algorithms. This I initially did not pay enough attention to until it was emphasized to me by Beresford Parlett (Parlett

(1987)). In order to use the divided-difference criterion (8) to bound the conditional average error (the left-hand-side of (9)), (8) has to be used to bound the difference between the approximation by this quadrature and another approximation given by the integral of the natural spline interpolation which is the mean of the conditional measure given the constraints $g(t_i) = f(t_i)$, $i=1, 2, \cdots, n$. Namely, to prove the theorem the following lemma was needed.

Lemma (Gao (1989))

$$\left|\int_a^b s_f(t)\,dt - \sum_{i=1}^{n} a_i f(t_i)\right| \leq \sqrt{c_2\, n}\,\frac{\bar{c}}{n^k} \max_{1\leq i\leq n} \left| [\, t_{i-k}, \ldots, t_i \,]\, f \right| \tag{10}$$

for any $(f(t_1), \cdots, f(t_n))^T \in R^n$ with the convention $f(t_i) = 0$, $i \leq 0$. Here s_f is the natural spline in $H_0^{k+1}[a,b]$ that interpolates f at $t_1, t_2, \cdots, t_n$.

Therefore, one can view an algorithm equipped with the error criterion (8) as one that uses the natural spline interpolants as the sample integrands and terminates successfully when the difference between the two solutions are small enough. This is in the spirit of the algorithms in practice where one often compares the approximation locally (in a sub-interval) with a different local approximation. The new algorithm does something stronger: it compares the approximate solution to a global one that has a minimal-norm property (see Gao (1989) for details), and yet the comparison is done implicitly, i.e., without having to compute the spline solution explicitly, and using only local error checks (divided differences, each depending on a small number of function values locally).

The main drawback again was the uniform partition condition (Condition 1) that excludes adaptive quadratures. In a later paper Gao (1990a), this point of view was further pursued, with the uniform partition condition relaxed, to derive new adaptive quadrature algorithms.

So, these results serve as an example to illustrate the value of studying computable error estimates in understanding practical numerical approximation algorithms. They are also in the spirit of Smale's general view on studying algorithms of numerical analysis from a computational complexity viewpoint expressed in, e.g., Smale (1985):

> Experience in the use of algorithms, especially with the computer in recent decades, has given rise to certain practices and beliefs. To give a deeper understanding of this culture, we try to give reasonable underlying mathematical formulations, and eventually to prove theorems, usually confirming the experience of the practitioners. Idealizations and simplifications are made, but we try to keep the essence of the observed phenomena. There is a kind of parallel in this approach to that of theoretical physics. Our primary goal is not the design of new algorithms, but we hope

that this deeper understanding will eventually be constructive in that domain too.

Compared to five years ago, today I am only more convinced of the practical and theoretical importance of studying computable error estimates to the design and analysis of numerical approximation algorithms, even though I have taken on other topics of interests in much of my research time. And I believe that I can now make better arguments for it based on the further reflections I have had.

A guaranteed error bound is important to evaluating a method of numerical approximation. As an *a priori* estimate, it provides information regarding the speed of convergence of the method and helps one decide whether the method is useful and where and when it should be used. A computable error estimate, on the other hand, is important as an error criterion in an algorithm of numerical approximation. As an *a posteriori* estimate, it allows the algorithm to decide whether a particular approximation is good enough, and if not what the next, more refined approximation should be. Therefore, the need to distinguish between the two types of error estimates is underlied by the need to distinguish between a method of numerical approximation -- an asymptotic way to approximate the exact solution of a problem -- and an algorithm of numerical approximation -- a finite process to construct one satisfactory approximation. While an algorithm is usually based on a method of approximation, it has other, more algorithmic aspects. A numerical approximation algorithm generates a sequence of approximations in order to obtain a satisfactory one. To do so, it has to choose an initial approximation, and after generating each approximation it has to determine whether the approximation is satisfactory and if not how to generate the next approximation. These more algorithmic aspects of a numerical approximation algorithm, namely the choice of initial approximation, the criterion for termination and the rules for adaption, often do not come with the method of approximation the algorithm uses. This latter distinction is important because it reveals the merits as well as limitations of traditional numerical analysis. Traditional numerical analysis has focused its study on guaranteed, *a priori* error bounds, and in doing so has produced a good understanding of many methods of numerical approximation. However, it has not given much study to computable, *a posteriori* error estimates that can be used in numerical approximation algorithms. This is not too surprising in hindsight since under the weak differentiability assumption most computable error estimates cannot be guaranteed error bounds; it would require more sophisticated mathematical models than the worst-case one traditional numerical analysis is accustomed to, to conduct a theoretical study. From a practical perspective, since the design of most of the error criteria in algorithms of numerical approximation have been heuristical, a theoretical study of computable error estimates can gain valuable insights to help improve existing algorithms and discover new algorithms. From a theoretical perspective, this is a terrain, relatively unexplored by traditional numerical analysis, where the mathematical analysis of numerical algorithms, with fresh ideas and novel tools, has the potential to establish itself and gain wider acceptance as a scientific field.

Acknowledgement

I wish to thank Jim Curry for the pointer to the paper by W. C. Rheinbolt.

References

Gao, F. (1986), Nonasymptotic error of numerical integration -- an average analysis, unpublished manuscript.

Gao, F. (1989), A probabilistic theory for error estimation in automatic integration, *Numerische Mathematik* 56, 309-329.

Gao, F. (1990a), On the power of *a posteriori* error estimation for numerical integration and function approximation, to appear in *Proc. AFA International Conference "Curves and Surfaces"* held in Chamonix, France.

Gao, F. (1990b), Probabilistic analysis of numerical integration algorithms, to appear in *Journal of Complexity.*

Lee, D., and Wasilkowski, G. W. (1986), Approximation of linear functionals on a Banach space with a Gaussian measure, *J. Complexity* **2,** 12-43.

Parlett, B. N. (1987), private communication.

Rheinbolt, W. C. (1988), On a theorem of S. Smale about Newton's method for analytic mappings, *Appl. Math. Lett.* Vol. 1, No. 1, 69-72.

Smale, S. (1985), On the efficiency of algorithms of analysis, *Bull. Amer. Math. Soc.* **13,** 87-121.

Smale, S. (1986), Newton's method estimates from data at one point, in *The Merging of Disciplines: New Directions in Pure, Applied and Computational Mathematics.* (R. E. Ewing, K. I. Gross, and C. F. Martin, eds.), Springer-Verlag, New York, 1986, 185-196.