1975

Remarks on Piecewise Polynomial Approximation

John R. Rice
Purdue University, jrr@cs.purdue.edu

Report Number:
75-163

REMARKS ON PIECEWISE POLYNOMIAL APPROXIMATION

John R. Rice
Purdue University
W. Lafayette, Indiana, U.S.A.

September 30, 1975

CSD TR 163

REMARKS ON PIECEWISE POLYNOMIAL APPROXIMATION

John R. Rice

Purdue University

W. Lafayette, Indiana, U.S.A.

Presented at the International Conference on the Approximation of Functions, Kalouga, SSSR, July 25, 1975

September 30, 1975

ABSTRACT

The theoretical and experimental background for the approximation of "real-world" functions is reviewed to motivate the use of piecewise polynomial approximations. It is possible in practice to achieve the results suggested by the theory for piecewise polynomials. A convergent adaptive algorithm is outlined which effectively and efficiently computes smooth piecewise polynomial approximations to any member of a broad class of functions. This class includes all functions which are piecewise smooth and have a finite number of "algebraic" singularities. Theoretically and experimentally determined properties of this algorithm are indicated.

REMARKS ON PIECEWISE POLYNOMIAL APPROXIMATION

John R. Rice*

1. THE EXPERIMENTAL BACKGROUND. The advent of high speed computers made it both possible and necessary to compute approximations to a large variety of functions. The functions involved are somewhat arbitrarily divided into two classes: the mathematical functions (e^x, $\Gamma(x)$, $J_1(x)$, etc.) and the "real world" functions (shape of a turbine blade or airplane wing, specific heat versus temperature for titanium, air pressure versus altitude, etc.) In this discussion we consider only the real-world functions and the remarks do not apply, in general, to the approximations of the elementary and special functions of mathematics. The experience of the 1950's was clear and convincing: classical linear methods of approximation are inadequate for real applications. We take ordinary polynomial approximations (say with L_2 or L_∞ norms) as the prime example of these classical methods. Other methods involving trigonometric functions (Fourier Series), Bessel functions, etc. are no better.

To make this position quantitative, we note that Rice made an experiment in the early 1960's as follows. Several dozen tabulated functions were selected at random from the "Handbook of Chemistry and Physics" which represent a variety of relationships in the real world. All of the

*This work was supported in part by grant GP 32940X from the National Science Foundation.

functions were approximated by polynomials of increasing degree until one of two things happened. First, the approximation achieved adequate accuracy to represent the physical process or, second, the computation broke down because the polynomial degree became too large. Note that straight forward computational methods using the powers of x as a basis for the polynomials broke down for very low degrees (6-10 on computers of that time). However, with a little thought one can compute with degrees up to 30 or 40 without difficulty. In only 50% of the cases could a satisfactory approximation be obtained by polynomials.

Some other method of approximation was required and two candidates were presented: rational approximation with the L_∞ norm and spline interpolation. Both of these candidates are still viable for this application although the spline ideas are currently further developed in practice and perhaps in theory. We do not discuss rational approximations further in this paper.

Spline interpolation was first done with equally spaced knots but soon some people observed that variable spaced knots improve things very much. Furthermore, it was realized that splines are only a particularly interesting case of general piecewise polynomials. From this evolves the nonlinear approximation problem of determining optimal or best knots for piecewise polynomial approximation. By now some algorithms exist for this problem, the oldest and

still widely used algorithm is that of deBoor and Rice [8], [9] which computes best least squares cubic spline approximations.

This algroithm has not, to the authors knowledge, failed to obtain a satisfactory approximation for real world functions. Note that the case of C^0 piecewise polynomials is particularly easy. One can think of algorithms, prove they converge and implement them on computers without difficulty. These are quite useful in many situations, but, unfortunately one often needs smoother approximations.

2. THE THEORETICAL BACKGROUND. The classical degree of convergence result for polynomial approximations may be stated as follows.

> THEOREM 1: Let $f(x)$ have an $(r-1)$st derivative which satisfies a Lipshitz condition. Further let P_n denote the polynomials of degree n or less. Then the distance in the L_p-norm satisfies
>
> $$\text{dist}_p(f,P_n) = \mathcal{O}(n^{-r})$$

This result indicates how the approximation error for a function $f(x)$ of smoothness r should behave as the degree n increases.

Early work on the degree of approximation by splines was done by Birkhoff and deBoor [1], Meir and Sharma [11] and others and it culminated in the following result. We

let S_k^r denote piecewise polynomials of order r(degree $\leq$ r-1) and with k knots; the case of splines comes from also assuming that the (r-2)nd derivative of the piecewise polynomial is continuous.

THEOREM 2: Let f(x) have an absolutely continuous (r-1)st derivative with $f^{(r)}(x)$ in $L_p[0,1]$. Let U_k^r be the subset of S_k^r of splines with uniformly spaced knots (the spacing is $1/k = |\pi|$). Then we have

$$\text{dist}_p(f,U_k^r) = \mathcal{O}(|\pi|^r) = \mathcal{O}(k^{-r})$$

This result also indicates how the approximation error for a function of smoothness r should behave as the number of knots increase. We see that the number k of knots in Theorem 2 plays the same role as the degree n of the polynomials in Theorem 1.

We are, however, primarily interested in results for optimal knots and the first result in this direction was by Rice [13] as follows. Let f(x) have singularities at the points s_i, i = 1, 2, ... m and set

$$w(x) = \prod_{i=1}^{m} (x-s_i)$$

THEOREM 3: Let f(x) be piecewise in $C^{r-1}(s_i,s_{i+1})$ and be in Lip(α) for some $\alpha > -1/p$. Further assume that there is a constant k so that

$$f^{(r-1)}(x) \leq K[w(x)]^{\alpha-r+1}$$

for $x \neq s_i$. Then we have

$$\text{dist}_p(f, S_k^r) = \mathcal{O}(k^{-r})$$

where the knots of S_k^r are variable. The imposition of smoothness requirements on elements of S_k^r does not change the result.

The important point, of course, is that the weakening the hypothesis on smoothness is completely compensated by allowing variable knots and the conclusion remains the same as for Theorem 2.

The functions involved in Theorem 3 include almost all (if not all) functions that arise in the real world. However, an examination of the proof of Rice immediately shows that functions with an infinite number of singularities can be included by this method. This raises the important question of just what functions are included and this question has been answered by the deep and difficult results (independently of one another) of Brudny: [2], Burchard [4], [5] and Peetre [12]. These results involve rather complicated technical constructions whose implications are not transparent, so we give instead a weaker and earlier result of Burchard [13], deBoor [6] and Dodson [10] where it is easier to see the nature of the functions involved. For given values of p and r let

$$\sigma = \frac{1}{r+1/p}$$

and define the σ-norm of $f(x)$ as

$$\| f \|_\sigma = [\int_0^1 |f^{(r)}|^\sigma dx]^{1/\sigma}$$

We may summarize the results by

THEOREM 4. Suppose $\| f \|_\sigma < \infty$ and $|f^{(r)}(x)|$ is monotone decreasing in some neighborhood of each singularity as one moves away from the singularity. Then we have

$$\text{dist}_p(f, S_k^r) = \mathcal{O}(k^{-r})$$

The σ-norm is something that one can calculate for many interesting simple functions and one sees, for example, that Theorem 4 includes Theorem 3 as a special case. Note that the monotone condition on $f^{(r)}(x)$ in Theorem 4 may be replaced by the assumption of a monotone bound such as seen in Theorem 3.

We also note that we have related the degree of the polynomial pieces to the smoothness of $f(x)$ in these results. The degree may be increased and the same degree of convergence is obtained.

3. THE PRACTICALITY OF COMPUTING APPROXIMATIONS. For applications one needs an approximation method with the following properties. It is efficient and reliable. It gives smooth approximation if desired. It gives approximations whose complexity is consistant with that of the function being approximated and the accuracy desired. The

results on degree of convergence given above suggest that this should be possible for polynomial or piecewise polynomial approximation. We ask the question: Is it possible to achieve the results suggested by the theory. The answer for polynomials is no. The essential failure occurs on the complexity issue; there are simple smooth functions where the polynomial degree must be huge (thousands or millions) to achieve approximations of modest accuracy (say 10^{-2}, 10^{-4} or 10^{-6}). It is difficult to compute such approximations as the computation is ill conditioned, but the results obtained are, in any case, useless. On the other hand, the answer for piecewise polynomials is yes. The next two sections describe how such computations may be made.

4. AN ADAPTIVE INTERVAL PARTITION ALGORITHM. The word adaptive means that the algorithm automatically adjusts itself to accomodate the special nature of the problem at hand. The adaptive approximation algorithm discussed in the next section depends heavily on an interval partition algorithm [14] originally introduced for quadrature.

PARTITION ALGORITHM:

1. Initialization: We are given

   A. Numbers γ, $\beta<1$ and $\varepsilon>0$.

   B. An empty set M' and a set M of intervals I with associated numbers $\eta(I)$. M contains a distinguished interval I*.

   C. A process $P:I \to (IL,IR)$ which divides an interval I

into left and right subintervals such that

(i) If $I = I^*$ then $\eta(IL) = \eta(IR) = \beta^*\eta(I)$ and $I^* \leftarrow IL$ or $I^* \leftarrow IR$.

(ii) If $I \neq I^*$ then $\eta(IL) = \eta(IR) = \gamma^*\eta(I)$

2. Operation:

For $I \in M$ do

$P: I \rightarrow (IL, IR)$

If $(\eta(IL) < \varepsilon)$ then $IL \in M'$ else $IL \in M$

If $(\eta(IR) < \varepsilon)$ then $IR \in M'$ else $IR \in M$.

We interpret this algorithm for approximation as follows: The interval [0,1] of approximation is being partitioned into subintervals and local polynomial approximations are being computed on them. The variables in the algorithm are:

$\eta(I)$ = approximation error on the subinterval I

I^* = the interval containing the singularity of $f(x)$.

ε = the desired accuracy or a quantity closely related to it.

γ = the reduction factor in the approximation error resulting from halving an interval which does not contain the singularity.

β = the error reduction factor for the interval I^* with the singularity.

In most applications one has $\gamma << \beta$.

This algorithm generates a tree of intervals as illustrated in Figure 1.

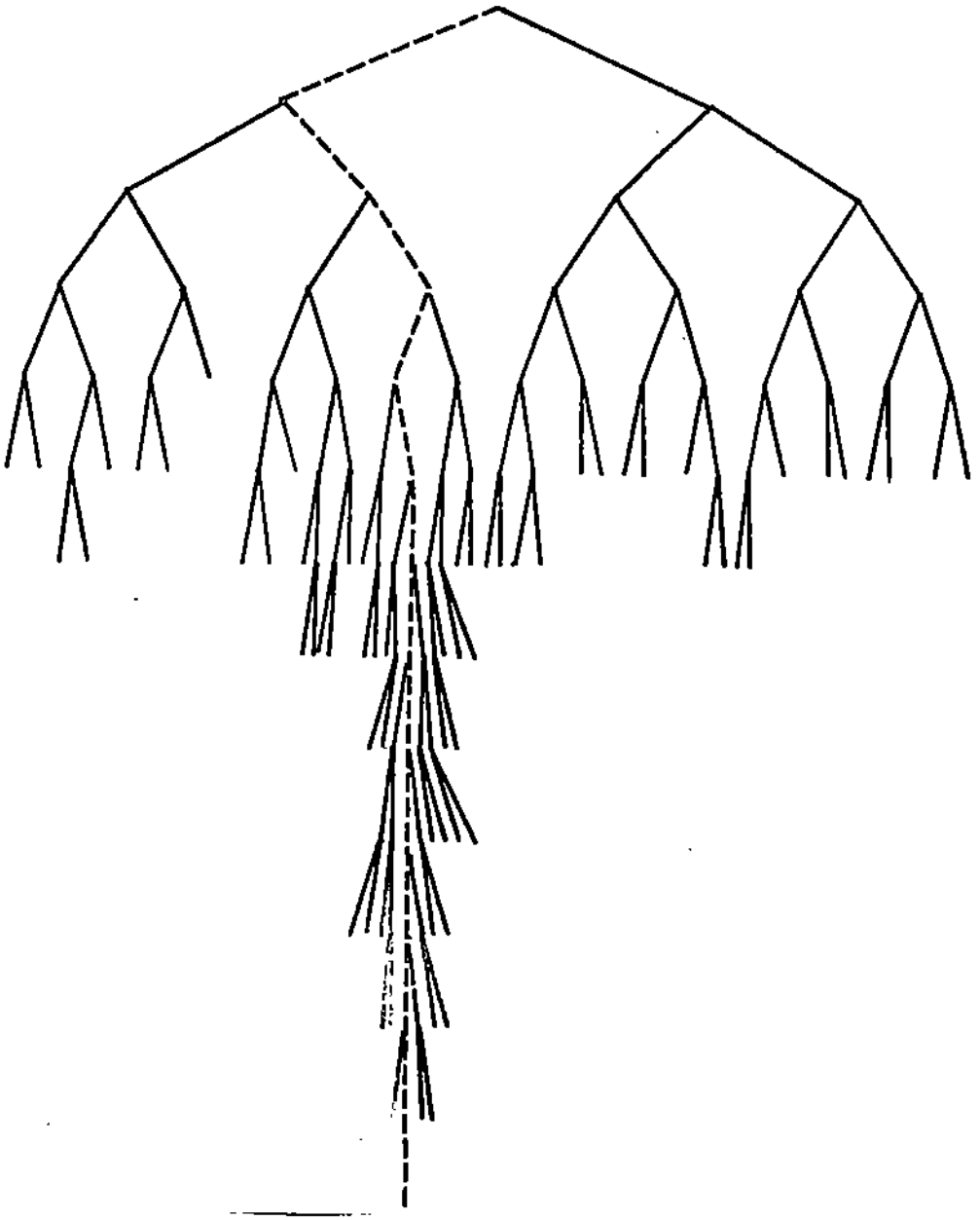

Figure 1. The tree of intervals generated by the partition algorithm. The dashed path is that of the distinguished interval.

The branches of the tree are followed until the corresponding interval can be discarded (i.e. the approximation is acceptable in that interval). Naturally the branches involving I* (i.e. intervals near the singularity) are much larger than the others. The key question is: how much longer are they and what proportion of the entire tree are they? This is answered by the following.

THEOREM 5. Consider the Partition Algorithm with β, M and $\eta(I)$ for $I \in M$ specified. Let $F(\gamma,\varepsilon)$ be the size of M' when the algorithm terminates and then we have

$$F(\gamma,\varepsilon) = \mathcal{O}(\varepsilon^{\frac{1}{\log_2 \gamma}})$$

There are a number of variations and extensions of this result given in [14], one example is

COROLLARY. Consider a real valued function g defined on intervals with the property that $I_1 \subseteq I_2$ implies $g(I_1) \leq g(I_2)$. Suppose that in the interval division process P the factors γ and β are replaced by $\gamma * g(IR)$, $\gamma * g(IL)$, $\beta * g(IR)$ and $\beta * g(IL)$ as appropriate. Then the conclusion of Theorem 5 remains valid.

The key point is that the result of theorem 5 is independent of β and hence the presence of a singularity does not affect the order of the overall amount of computation required. The quantity in the conclusion of Theorem 5 is

just the number of nodes of the tree if it were completely uniform, that is, all the paths from the apex were the same length.

5. APPLICATION TO PIECEWISE POLYNOMIAL APPROXIMATION. An adaptive approximation method involves a local approximation operator T_I which associates with $f(x)$ an approximation $A_I(f,x)$ on the interval I, i.e.

$$T_I : f(x) \to A_I(f,x)$$

A simple example of this is linear interpolation at the end points of the interval I. Associated with the adaptive method is a tolerance $\varepsilon > 0$ and an interval I is discarded if

$$\| f(x) - A_I(f,x) \|_I = \| (1-T_I)f \|_I < \varepsilon$$

The subscript on the norm indicates restriction to the interval I. The number ε is not necessarily the desired approximation accuracy.

When the adaptive method terminates, we know that the local error of approximation on each interval is less than ε and this allows one to estimate the global error depending on the nature of the norm used. The global approximation is, of course, just the collection of local approximations $A_I(f,x)$. For simplicity, we assume that each interval is halved and thus each interval is of the form $[x,x+2^{-j}]$ for some value of j and we may represent it by the pair (x,j). For specific local approximation operators and suitable

functions $f(x)$ we have a bound on the error $\|(1-T_I)f\|_I$ and we denote this by ERROR(x,j).

We consider functions such as those involved in Theorem 3 above. We have

ASSUMPTION 1. Assume $f(x)$ has singularities

$$S = \{s_i \mid i = 1,2,\ldots,m < \infty\}$$

and set

$$w(x) = \prod_{i=1}^{m} (x-s_i).$$

(i) If $x_0 \notin S$ then $f^{(r)}(x)$ is continuous in a neighborhood of x_0 with $r \geq 1$.

(ii) There are constants K and α so that

$$|f^{(r)}(x)| \leq K|w(x)|^{\alpha-r}$$

We consider local approximation operators T_I which satisfy:

ASSUMPTION 2. Let s denote a point of singularity of $f(x)$ and set

$$F_r(x,j) = \max_{t\in[x,x+2^{-j}]} |f^{(r)}(t)|$$

There are constants r, K and α (the same as in Assumption 1) so that:

(i) $\text{ERROR}(x,j) \leq KF_r(x,k)2^{-j(r+1/p)}$ if $[x,x+2^{-j}]$ contains no singularity

(ii) $\text{ERROR}(x,j) \leq K2^{-(\alpha+1/p)j}$ if $s\in[x,x+2^{-j}]$

The rate of convergence of this algorithm has been determined by Rice [15] as follows.

THEOREM 6. Let $f(x)$ satisfy Assumption 1 with $\alpha > -1/p$ for $1 \leq p \leq \infty$. Consider an adaptive algorithm whose local operator satisfies Assumption 2. Then the L_p-error of the global approximation $A(x)$ obtained when the algorithm terminates is of the order

$$\mathcal{O}(k^{-r})$$

where k is the number of pieces of $A(x)$.

This result states that an adaptive algorithm achieves the optimal rate of convergences provided the local approximation operators are suitably chosen.

There are two classes of operators which are known to satisfy the assumptions of Theorem 6. These are the local Hermite interpolation operators and the spline quasi-interpolant operators of deBoor and Fix [7]. The Hermite operators are the simplest to analyze and use and they have been incorporated into an actual algorithm (computer program) for adaptive piecewise polynomical approximations [16]. The Hermite operators have the disadvantage that the polynomial degree is at least twice the number of continuous derivatives of the approximation. We present a brief outline of the resulting algorithm in a very high level programming language which is hopefully self-explanatory.

```
   F       · FUNCTION TO FIT        SMOOTH · NO. CONT. DERIVS
   A,B     · INTERVAL ENDPTS        NORM   · MEAS. OF L-P ERROR
   ACCUR   · ACCURACY DESIRED                P IN (0,INFINITY)
   DEGREE  · POLYNOMIAL DEGREE

PROGRAM ADAPT

   *** LOOP OVER PROCESSING INTERVALS ***

   CALL TAKE   - AN INTERVAL OFF THE STACK
   CALL COMPUT - AN APPROX ON THIS INTERVAL
   CALL CHECK  - FOR DISCARDING OR DIVIDING INTERVAL
   CALL PUT    - NEW INTERVALS ON STACK, UPDATE ALGORITHM STATUS
   CALL TERMIN - TEST FOR FINISH, PRINT INTERMEDIATE OUTPUT
   IF NOT FINISHED - REPEAT LOOP

END ADAPT

SUBPROGRAM TAKE
   MAKE THE TOP INTERVAL OF THE STACK AVAILABLE
END TAKE

SUBPROGRAM COMPUT
   OBTAIN      - VALUES OF F AND DERVIATIVES.
   CALL NEWTON - FOR DIVIDED DIFFERENCES OF INTERPOLATING
                 POLYNOMIAL FOR THIS INTERVAL
   CALL ERRINT - TO ESTIMATE LOCAL ERROR IN L-P NORM
END COMPUTE

   SUBPROGRAM NEWTON - OF COMPUT
       BUILD UP TRUE DIVIDED DIFFERENCE TABLE WITH MULTIPLE
       POINTS AT THE INTERVAL ENDS PLUS INTERPOLATION POINTS
   END NEWTON

   SUBPROGRAM ERRINT - OF COMPUT
       USES 4-POINT GAUSS QUADRATURE TO ESTIMATE ERROR NORM ON
       INTERVAL. SPECIAL COMPUTATION FOR MAX-NORM, P = INFINITY.
   END ERRINT

SUBPROGRAM CHECK
   MAKE DECISION ON DISCARDING THE INTERVAL
END CHECK

SUBPROGRAM PUT

   CHECK FOR DISCARDING INTERVAL
       IF SO - UPDATE ERROR ESTIMATE
               ADJUST STACK
               CALL PTRANS - TO OBTAIN COEFS FOR THIS INTERVAL
               UPDATE XKNOTS AND COEFS

       ELSE - SUBDIVIDE INTERVAL AND PLACE 2 NEW ONES ON STACK
END PUT
```

```
   SUBPROGRAM PTRANS - OF PUT
       CHANGES POLYNOMIAL REPRESENTATION FROM NEWTON DIVIDED
       DIFFERENCE FORM TO POWER FORM WITH ORIGIN SHIFTED TO
       THE XKNOT VALUE ON LEFT OF INTERVAL. USES SYNTHETIC
       DIVISION
   END PTRANS

SUBPROGRAM TERMIN
   STOP WHEN THE STACK IS EMPTY.
END TERMIN
```

This algorithm has been extensively tested to see if it does achieve the results suggested by theory. Part of this work is reported upon in [17] and these tests justify the statement that it does perform as hoped. A number of other properties are observed in these tests and mentioned in [17]

REFERENCES

1. Birkhoff, G. and deBoor, C., Piecewise polynomial interpolation and approximation, in "Approximation of Functions" (H. L. Garabedian, ed.), Elsevier Pub. Co., New York, 1965, pp. 164-190.

2. Brudnyi, A., Spline approximation and functions of bounded variations, Dolk. Acad. Nauk SSSR, 215 (1974) pp. 511-514

3. Burchard, H. G., Splines (with optimal knots) are better, J. Applicable Analysis, 3 (1974) pp. 309-319

4. _____, On the degree of convergence of piecewise polynomial approximation on optimal meshes II,

5. _____and Hale, D. F., Piecewise polynomial approximation on optimal meshes, J. Approx. Thy, (1975)

6. deBoor, C. W., Good approximation by splines with variable knots, in "Spline Functions and Approximation Theory" (A. Meir and A. Sharman, eds.), ISNM 21, Birkhauser Verlag, Basel (1973) pp 57-72.

7. _____ and Fix, G., Spline approximation by quasi-interpolants, J. Approx. Thy. 7 (1973), 19-45.

8. _____and Rice, J. R. Least squares cubic spline approximation I: Fixed knots, CSDTR 20, Purdue Univ. 1968

9. _____, Least squares cubic spline approximation II: Variable knots: CSDTR 21, Purdue Univ. 1968

10. Dodson, D. S., Optimal order approximation by polynomial spline functions, Thesis, Purdue University, 1972.

11. Meir, A. and Sharma, A., Degree of approximation of spline interpolation, J. Math., Mech., 15 (1966) pp. 759-767.

12. Peetre, J. and Bergh, J., On the spaces $V_p (0 < p \leq \infty)$, to appear

13. Rice, J. R., On the degree of convergence of nonlinear spline approximation, in "Approximations with special emphasis on spline functions", (I.J. Schoenberg ed.), Academic Press, New York (1969), 349-365.

14. _____. A metalgorithm for adaptive quadrature, J. Assoc. Comp. Mach., 22 (1975) pp. 61-82.

15. _____, Adaptive approximation, J. Approx. Thy., to appear.

16. _____, ADAPT-Adaptive smooth curve fitting, to appear.

17. _____, On adaptive piecewise polynomial approximation, in "Theory of Approximation" (A. Law and B. Sahney, eds.) Academic Press, New York (1976)