ALGORITHM 604
A FORTRAN Program for the Calculation of an Extremal Polynomial

FREDERICK W. SAUER
MRC, University of Wisconsin

Categories and Subject Descriptors: G.1.2 [**Numerical Analysis**] Approximation—*minimax approximation and algorithms*; G.1.3 [**Numerical Analysis**]: Numerical Linear Algebra—*linear systems (direct and iterative methods)*, G.m [**Mathematics of Computing**]. Miscellaneous—*FORTRAN*

General Terms: Algorithms

Additional Key Words and Phrases: Remes, extremal polynomial, Richardson iteration

DESCRIPTION

Let S be the union of finitely many compact intervals in $\mathbf{R}$ such that $S \cap (-\infty, 0] \neq \emptyset$, $S \cap [0, +\infty) \neq \emptyset$, and $0 \notin S$. Let λ be the linear functional on π_n (:= the polynomials of degree n or less) whose value at p is $p(0)$. An *extremal* for λ is any polynomial of norm 1 at which λ takes on its norm, that is, $p \in \pi_n$, $\|p\|_S = 1$, and $\lambda p = \|\lambda\|$. Here

$$\|p\|_S := \max_{s \in S} |p(s)|$$

and

$$\|\lambda\| := \sup_{p \in \pi_n, \|p\|_S = 1} |\lambda p| = 1/\min\{\|p\|_S : p \in \pi_n, p(0) = 1\}.$$

In [1] it is proved that for a given S and n there exists a unique extremal polynomial. The program presented here is a FORTRAN implementation of the Remes algorithm outlined in [1] to calculate the extremal polynomial given S and n.

The extremal polynomial can be used for the determination of the iteration parameters α_n for Richardson's iteration

$$\mathbf{x}^{n+1} = \mathbf{x}^n - \alpha_n (A\mathbf{x}^n - \mathbf{b})$$

Received 11 November 1980; revised 15 December 1982; accepted 29 December 1982
This work was sponsored by the U.S. Army under Contract DAAG29-80-C-0041.
Author's address, Mathematics Research Center, University of Wisconsin, 610 Walnut Street, Madison, WI 53706.

to solve a linear system of equations $A\mathbf{x} = \mathbf{b}$. Defining $\mathbf{e}^n := \mathbf{x}^n - \mathbf{x}$, the error in the nth iterate, we have

$$\mathbf{e}^n = (1 - \alpha_{n-1}A)\mathbf{e}^{n-1} = \cdots = \prod_{j=1}^{n} (1 - \alpha_{j-1}A)\mathbf{e}^0 = Q_n(A)\mathbf{e}^0,$$

where Q_n is the polynomial of degree n, which vanishes at $1/\alpha_0, \ldots, 1/\alpha_{n-1}$ and is 1 at 0. If the spectrum of A is known to lie in some compact set S, then one should choose the parameters so as to minimize $\| Q_n \|_S$; that is, one should choose the parameters to be the inverse of the roots of the extremal polynomial for n and S. The resulting polynomial P_n is then the error in the *best Chebyshev approximation on S from* $\{\sum_{j=1}^{n} \beta_j t^j\}$.

A description of how to use this program is given in the comments in subroutine **EXTREM**. In addition, two sample programs are provided. The first is a simple example how **EXTREM** can be called. The second uses Richardson's iteration to solve the linear system $(B^2 - \mu I)x = b$, with $\mu = \sqrt{3}$ and B the tridiagonal matrix with 2 on the diagonal and -1 on the off-diagonals.

REFERENCES

1. DE BOOR, C., AND RICE, J.R. Extremal polynomials with applications to Richardson iteration for indefinite linear systems. *SIAM J. Sci. Stat. Comput. 3*, 1 (1982), 47–57.

ALGORITHM

[A part of the listing is printed here. The complete listing is available from the ACM Distribution Service (see page 385 for order form).]

```
      SUBROUTINE EXTREM(X, N, XINT, NINT, K, IPRINT, MAXIT, EPS2, EPS3,
     * EPS4, XTILDE, ALPHA, PROD, PZERO, IERR)
C THIS SUBROUTINE FINDS A POLYNOMIAL WHICH IS AN EXTREMAL OF THE LINEAR
C FUNCTIONAL WHICH IS THE POINT EVALUATION AT THE POINT ZERO.  THE
C POLYNOMIAL IS CHOSEN FROM THE SPACE OF POLYNOMIALS OF DEGREE LESS
C THAN OR EQUAL TO N, WHERE N IS A PARAMETER SET BY THE USER. THE NORM
C ON THIS SPACE IS DEFINED TO BE:
C
C       NORM(P) = MAX(ABS(P(T)) : T IS IN S)
C
C WHERE S IS THE UNION OF THE CLOSED INTERVALS (XINT(1,I),XINT(2,I)),
C I = 1, 2, ..., NINT.
C
C THE REMES ALGORITHM (OUTLINED BELOW) IS USED TO FIND THIS EXTREMAL
C  POLYNOMIAL.
C
C
C                          PARAMETERS
C
C X        A REAL ARRAY OF DIMENSION N+1. ON INPUT, X CONTAINS A
C          STRICTLY INCREASING SEQUENCE OF POINTS IN S WITH X(K)=B AND
C          X(K+1)=C, WHERE B AND C ARE DEFINED:
C             B = MAX (T : T IS IN S, AND T .LE. 0)
C             C = MIN (T : T IS IN S, AND T .GE. 0).
C          ON RETURN, X CONTAINS THE FINAL SEQUENCE WHICH DEFINES THE
C          EXTREMAL POLYNOMIAL WHICH IS THE UNIQUE POLYNOMIAL SUCH THAT:
C
C                        (-1)**(K-J)    IF J .LE. K
C            P(X(J)) =
C                        (-1)**(J+1+K)  IF J .GT. K.
C
C          A SUITABLE STARTING SEQUENCE FOR X CAN BE OBTAINED BY
C          CALLING SUBROUTINE SETUP.
```

```
C  N        DEGREE OF EXTREMAL POLYNOMIAL. IF THIS IS TOO LARGE
C           OVERFLOW, UNDERFLOW, OR DIVISION BY ZERO ARE LIKELY
C  XINT     TWO DIMENSIONAL ARRAY OF DIMENSION 2 BY NINT. ON INPUT, XINT
C           CONTAINS THE ENDPOINTS OF THE INTERVALS DEFINING S (SEE ABOVE
C           DEFINITION OF S). XINT MUST BE IN INCREASING ORDER:
C               XINT(1,1) .LE. XINT(2,1) .LT. XINT(1,2) .LE. XINT(2,2) ..
C               .LT. XINT(1,NINT) .LE. XINT(2,NINT).
C           XINT MUST BE SUCH THAT S CONTAINS AT LEAST N+1 POINTS.
C  NINT     NUMBER OF INTERVALS IN S.
C  K        INTEGER SUCH THAT X(K)=B AND X(K+1)=C.
C  IPRINT   INTEGER WHICH ON INPUT HAS THE FOLLOWING MEANINGS
C             =-1 SUPPRESS ALL PRINTING. PZERO WILL NOT BE CALCULATED
C             =0  PZERO = P(0) WILL BE CALCULATED AND PRINTED
C             =1  PZERO AND THE ROOTS OF THE POLYNOMIAL WILL BE
C                     CALCULATED AND PRINTED
C             =2  IN ADDITION, FOR EACH ITERATION THE CURRENT VALUE OF
C                     THE ARRAY X WILL BE PRINTED
C  MAXIT    MAXIMUM NUMBER OF ITERATIONS ALLOWED
C  EPS2     A TOLERANCE USED AS A STOPPING CRITERION FOR THE MAJOR
C           ITERATION. THE ROUTINE WILL STOP WHEN
C              MAX(ABS(XOLD(I)-X(I)),I=1,2,...,N+1) .LT. EPS2
C           WHERE XOLD IS THE X ARRAY OF THE PREVIOUS ITERATION.
C  EPS3,EPS4  TOLERANCE PARAMETERS USED BY SUBROUTINE REGULA. THE
C           CALCULATED VALUE OF THE ROOT, XRT, WILL BE SUCH THAT XRT WILL
C           BE WITHIN EPS3 OF AN ACTUAL ROOT OR ABS(P(XRT)) WILL BE LESS
C           THAN EPS4. VALUES FOR EPS3 AND EPS4 NEED NOT BE PROVIDED IF
C           IPRINT IS EQUAL TO -1 OR 0.
C  XTILDE   ARRAY OF DIMENSION N+1. ON OUTPUT, IF IPRINT IS EQUAL TO 1 OR
C           2, XTILDE WILL CONTAIN THE RECIPROCALS OF THE ROOTS OF THE
C           EXTREMAL POLYNOMIAL. THE RECIPROCALS OF THE ROOTS IN THE
C           INTERVAL (X(1),X(N+1)) WILL BE STORED IN THE ELEMENTS
C           XTILDE(2),XTILDE(3),...,XTILDE(N). THE RECIPROCAL OF THE
C           REMAINING ROOT, IF IT EXISTS, WILL BE STORED IN XTILDE(1).
C           IF IT DOES NOT EXIST, XTILDE(1) WILL BE SET TO ZERO.
C  ALPHA    ARRAY OF DIMENSION N+1 USED BY THIS ROUTINE TO STORE THE
C           VALUES (AS CALCULATED BY ALCALC)
C
C                                       N+1
C          ALPHA(I) =  P(X(I))  / PRODUCT   (X(I)-X(J))
C                                       J=1
C                                     J.NE.I
C
C          WHERE P(X) IS THE VALUE OF THE POLYNOMIAL AT X. IN PARTICULAR,
C
C                          (-1)**(I-K)    IF I .LE. K
C              P(X(I))=
C                          (-1)**(I+1-K)  IF I .GT. K
C
C           ON RETURN, IF IPRINT IS UNEQUAL TO -1, ALPHA WILL CONTAIN
C           THESE VALUES FOR THE EXTREMAL POLYNOMIAL.
C
C  PROD     ARRAY OF DIMENSION N+1 USED BY THIS ROUTINE TO STORE
C           THE VALUES (AS CALCULATED BY ALCALC)
C
C                           N+1
C          PROD(I) = PRODUCT (X(I) - X(J))
C                           J=1
C                          J.NE.I
C
C           ON RETURN, IF IPRINT IS UNEQUAL TO -1, PROD WILL CONTAIN
C           THESE VALUES FOR THE EXTREMAL POLYNOMIAL.
C
C  PZERO    ON OUTPUT, IF IPRINT IS UNEQUAL TO -1, PZERO WILL CONTAIN
C           THE VALUE OF THE EXTREMAL POLYNOMIAL AT 0.
C
C  IERR     ON RETURN, IERR WILL BE THE SUM OF THE VALUES OF THE
C           FOLLOWING POSSIBLE USER INPUT ERRORS.  IERR=0 IS THE
C           NORMAL RETURN.  IF IERR IS NOT ZERO THE ROUTINE WILL
C           RETURN IMMEDIATELY WITHOUT CALCULATING THE EXTREMAL
C           POLYNOMIAL
C
```