APPROXIMATION THEORY V

Edited by C.K. Chui, L.L. Schumaker, and J.D. Ward
Center for Approximation Theory
Department of Mathematics
Texas A&M University
College Station, Texas

Academic Press, Inc.
Harcourt Brace Jovanovich, Publishers

Boston Orlando San Diego
New York Austin London Sydney
Tokyo Toronto

MATLAB PROGRAMS FOR CF APPROXIMATION

Lloyd N. Trefethen

Programs for real and complex rational CF (Carathéodory-Fejér) approximation are presented for the interactive matrix calculator MATLAB.

1. Introduction

CF approximation (≈ AAK approximation [1]) is an analytical procedure for near-best rational approximation of real or complex functions in the supremum norm. Its distinguishing feature is the use of a singular value decomposition of a Hankel matrix of Taylor coefficients (complex case) or Chebyshev coefficients (real case). CF approximants offer a practical alternative to Chebyshev approximants, which are often difficult to compute, especially in complex domains, where they are not even unique.

This paper offers a pair of MATLAB programs for the computation of CF approximants. MATLAB is an advanced interactive matrix calculator, designed by Cleve Moler and others in the last decade, that provides convenient access to most of the capabilities of EISPACK and LINPACK, together with many additional features. (For information, contact The MathWorks, Inc., 158 Woodland St., Sherborn, MA 01770, tel. (617) 653-1415.) MATLAB's power, simplicity, and elegance make it a joy to use — an irresistible tool for a mathematician with an experimental bent.

CF approximation requires a sequence of linear algebra and Fourier transform calculations that appears daunting when written in a standard programming language. The higher-level language of MATLAB commands makes possible a much more natural specification. These two programs are offered to communicate the CF idea with brevity and precision, for small-scale experimentation, and as benchmarks for the design of faster and more foolproof CF programs of the usual kind.

2. Complex CF approximation on $|z| \le 1$ [2]

Let $f(z)$ be a complex analytic function on $|z| \le 1$, and let fz be a MATLAB function that calculates it. The MATLAB program CF of Figure 1 computes the complex CF approximant $r(z) = p_m(z)/q_n(z)$ of type (m,n). To run CF, one provides a value nfft specifying how long an FFT will be used in computing Taylor coefficients (e.g. 32 for a smooth function, 1024 for a troublesome one), and an integer K specifying at what Taylor series index the Hankel matrix will be truncated (e.g. 10 or 50).

For a very simple example, a run of CF with $f(z) = \sqrt{1.2 - z}$ produced the following results in about 40 seconds on an IBM PC/AT. (The output is condensed to save space.)

ISBN 0-12-174581-3

```
% CF -- COMPLEX RATIONAL CF APPROXIMATION ON THE UNIT DISK
%
% Lloyd N. Trefethen, Dept. of Math., M.I.T., March 1986
% Reference: L.N.T., Numer. Math. 37 (1981), pp. 297-320
% (This is "Type 2" approx., slightly different from above.)
%
%    fz(z) - function to be approximated by r(z)=p(z)/q(z)
%      m,n - degree of p,q
%     nfft - number of points in FFT (power of 2)
%        K - degree at which Taylor series is truncated
%  f,p,q,r - functions evaluated on FFT mesh (roots of unity)
%    pc,qc - coefficients of p and q
%
% If fz is even, take (m,n) = ( odd,even).
% If fz is  odd, take (m,n) = (even,even).
%
% If fz has complex Taylor coefficients, delete the "real"
% commands below.
%
% CONTROL PARAMETERS
    m = input('m? '); n = input('n? '); mp = m+1; np = n+1;
    nfft = input('nfft? ');
    K = input('K? '); dim = K+n-m;
%
% TAYLOR COEFFICIENTS OF fz
    z = exp(2*pi*sqrt(-1)*(0:nfft-1)/nfft);
    f = fz(z); fc = real(fft(f))/nfft;
    fc(nfft/2+1:nfft) = zeros(1,nfft/2);
%
% SVD OF HANKEL MATRIX H
    H = toeplitz(fc(1+rem((dim:-1:1)+nfft+m-n,nfft)));
      H = triu(H); H = H(:,(dim:-1:1));
    [u,s,v] = svd(H);
    s = s(np,np); u = u((dim:-1:1),np)'; v = v(:,np)';
%
% DENOMINATOR POLYNOMIAL q
    zr = roots(v); qout = []; for i = 1:dim-1;
      if abs(zr(i))>1 qout = [qout, zr(i)]; end; end;
    qc = real(poly(qout)); qc = qc/qc(np); q = polyval(qc,z);
%
% NUMERATOR POLYNOMIAL p
    b = fft([u zeros(1,nfft-dim)])./fft([v zeros(1,nfft-dim)]);
    rt = f-s*z.^K.*b; rtc = real(fft(rt))/nfft;
    pc = conv(qc(np:-1:1),rtc(1:mp)); pc = pc(mp:-1:1);
    p = polyval(pc,z); r = p./q;
%
% RESULTS
    axis('square'); plot([f-r f(1)-r(1)]); pause;
    s, err = norm(f-r,'inf'), pc, qc
```

Figure 1. MATLAB program for complex CF approximation

```
% RCF -- REAL RATIONAL CF APPROXIMATION ON THE UNIT INTERVAL
%
% Lloyd N. Trefethen, Dept. of Math., M.I.T., March 1986
% Reference: L.N.T. and M.H. Gutknecht,
%            SIAM J. Numer. Anal. 20 (1983), pp. 420-436
%
%    Fx(x) - function to be approximated by R(x)=P(x)/Q(x)
%      m,n - degree of P,Q
%     nfft - number of points in FFT (power of 2)
%        K - degree at which Chebyshev series is truncated
%  F,P,Q,R - functions evaluated on FFT mesh (Chebyshev points)
%    Pc,Qc - Chebyshev coefficients of P and Q
%
% If Fx is even, take (m,n) = ( odd,even).
% If Fx is  odd, take (m,n) = (even,even).
%
% CONTROL PARAMETERS
    m = input('m? '); n = input('n? '); np = n+1;
    nfft = input('nfft? '); nfft2 = nfft/2;
    K = input('K? '); dim = K+n-m;
%
% CHEBYSHEV COEFFICIENTS OF Fx
    z = exp(2*pi*sqrt(-1)*(0:nfft-1)/nfft);
    x = real(z); F = Fx(x); Fc = real(fft(F))/nfft2;
%
% SVD OF HANKEL MATRIX H
    H = toeplitz(Fc(1+rem((dim:-1:1)+nfft+m-n,nfft)));
      H = triu(H); H = H(:,(dim:-1:1));
    [u,s,v] = svd(H);
    s = s(np,np); u = u((dim:-1:1),np)'; v = v(:,np)';
%
% DENOMINATOR POLYNOMIAL Q
    zr = roots(v); qout = []; for i = 1:dim-1;
      if abs(zr(i))>1 qout = [qout, zr(i)]; end; end;
    qc = real(poly(qout)); qc = qc/qc(np); q = polyval(qc,z);
    Q = q.*conj(q); Qc = real(fft(Q))/nfft2;
    Qc(1) = Qc(1)/2; Q = Q/Qc(1); Qc = Qc(1:np)/Qc(1);
%
% NUMERATOR POLYNOMIAL P
    b = fft([u zeros(1,nfft-dim)])./fft([v zeros(1,nfft-dim)]);
    Rt = F-real(s*z.^K.*b); Rtc = real(fft(Rt))/nfft2;
    gam = real(fft((1)./Q))/nfft2; gam = toeplitz(gam(1:2*m+1));
    if m==0 Pc = 2*Rtc(1)/gam;
      else Pc = 2*[Rtc(m+1:-1:2) Rtc(1:m+1)]/gam; end;
    Pc = Pc(m+1:2*m+1); Pc(1) = Pc(1)/2;
    P = real(polyval(Pc(m+1:-1:1),z)); R = P./Q;
%
% RESULTS
    plot(x,F-R,'-', x,[s;0;-s]*ones(1,nfft),':'); pause;
    s, err = norm(F-R,'inf'), Pc, Qc
```

Figure 2. MATLAB program for real CF approximation

```
m? 1     n? 1     nfft? 128     K? 20
s = .03252        err = .03320
r(z) = (1.09497 - .74277z) / (1 - .26688z)
```

The singular value s is a lower bound for the error in best Chebyshev approximation, and err is the error of the near-best CF approximant. Thus in this case the CF result is within 3% of optimal. The penultimate line of the program CF generates a plot of the error curve $(f-r)(|z|=1)$, which comes out within a few percent of a circle of winding number 3.

3. Real CF approximation on $-1 \le x \le 1$ [3]

Now let $F(x)$ be a real function on $1 \le x \le 1$ calculated by a MATLAB function Fx. The MATLAB program RCF of Figure 2 computes the corresponding real CF approximant $R(x) = P_m(x)/Q_n(x)$ of type (m,n). Its operation is just like that of CF.

A run of RCF with the same function as before yields these results:

```
m? 1     n? 1     nfft? 128     K? 20
s = .0100706    err = .0100751
R(x) = (1.10417 - .77197x) / (1 - .27354x)
```

The error is smaller than before, since the same function is being approximated on a smaller domain. In real CF approximation, the singular value is no longer necessarily a lower bound for the error, but the plot generated by RCF reveals an error curve $(F-r)([-1,1])$ that equioscillates to within 0.1%, showing that the CF result is within this distance of optimal.

Acknowledgments

I am grateful to Martin Gutknecht for years of collaboration on CF approximation, to Alar Toomre for his PC/AT, and to the National Science Foundation for support under Grant DMS-8504703.

References

1. Adamjan, V., Arov, D., and Krein, M., Analytic properties of Schmidt pairs for a Hankel operator and the generalized Schur-Takagi problem, Math. USSR Sb., 15 (1971), 31-73.
2. Trefethen, L. N., Rational Chebyshev approximation on the unit disk, Numer. Math., 37 (1981), 297-320.
3. Trefethen, L. N. and Gutknecht, M. H., The Carathéodory-Fejér method for real rational approximation, SIAM J. Numer. Anal., 20 (1983), 420-436.

Lloyd N. Trefethen
Department of Mathematics
Massachusetts Institute of Technology
Cambridge, MA 02139