Sixteenth Midwest Symposium on Circuit Theory

University of Waterloo
Waterloo, Ontario, Canada
April 12 - 13, 1973

Proceedings, vol. 2

siam

TWO PROGRAMS FOR LEAST pTH APPROXIMATION

J.R. Popović, J.W. Bandler and C. Charalambous
Department of Electrical Engineering
McMaster University
Hamilton, Ontario, Canada.

Abstract

User-oriented computer programs in FORTRAN IV for discrete least pth approximation with a single specified function, and more generalized discrete least pth approximation with various specifications, which may also be used for nonlinear programming, are presented. Values of p up to 10^6 can be used successfully in conjunction with efficient gradient minimization algorithms such as the Fletcher-Powell method and a method due to Fletcher. The programs may be applied to a wide variety of design problems with a wide range of specifications. They are suitable for electrical network and system design and such problems as filter design.

INTRODUCTION

Two complete user-oriented computer programs in FORTRAN IV are presented which utilize some new ideas on discrete least pth approximation [1]. Least pth approximation with p=2 gives a discrete least squares approximation. With sufficiently large values of p an optimal solution very close to the optimal minimax solution can be obtained. Values of p up to 10^6 have been successfully employed. Gradient minimization algorithms due to Fletcher and Powell [2] and, more recently, to Fletcher [3] are used. The user has to write all the required specifications, the approximating functions and weighting functions in a straightforward way.

DEFINITIONS

Define real weighted error functions related to the upper and lower specifications, respectively, as [1]

$$e_u(\underset{\sim}{a},x) \triangleq w_u(x)(F(\underset{\sim}{a},x) - S_u(x)) \tag{1}$$

$$e'_u(\underset{\sim}{a},x,\xi) \triangleq w_u(x)(F(\underset{\sim}{a},x) - S'_u(x,\xi)) = e_u(\underset{\sim}{a},x) - \xi \tag{2}$$

$$e_\ell(\underset{\sim}{a},x) \triangleq w_\ell(x)(F(\underset{\sim}{a},x) - S_\ell(x)) \tag{3}$$

$$e'_\ell(\underset{\sim}{a},x,\xi) \triangleq w_\ell(x)(F(\underset{\sim}{a},x) - S'_\ell(x,\xi)) = e_\ell(\underset{\sim}{a},x) + \xi \tag{4}$$

This work was supported by the National Research Council of Canada under grants A7239 and C154. The programs and complete documentation are available from J.W. Bandler at a nominal charge.

where $F(\underset{\sim}{a},x)$ is the approximating function, $S_u(x)$ is the upper specified function, $S_u'(x,\xi)$ is the artificial upper specified function, $S_\ell(x)$ is the lower specified function, $S_\ell'(x,\xi)$ is the artificial lower specified function, $w_u(x)$ is the upper positive weighting function, $w_\ell(x)$ is the lower positive weighting function, $\underset{\sim}{a}$ is the vector containing the k independent parameters, x is an independent variable and ξ is a margin of errors with respect to the artificial and desired specifications.

When upper and lower specified functions and weighting functions coincide, respectively, let $S(x) = S_u(x) = S_\ell(x)$ and $w(x) = w_u(x) = w_\ell(x)$. Then from (1) and (3) $e(\underset{\sim}{a},x) = e_u(\underset{\sim}{a},x) = e_\ell(\underset{\sim}{a},x)$.

In practice we will evaluate all the functions at a finite discrete set of values of x taken from one or more closed intervals. Therefore, we will let

$$e_{ui}'(\underset{\sim}{a},\xi) \triangleq e_u'(\underset{\sim}{a},x_i,\xi) \quad , \quad i\in I_u \tag{5}$$

$$e_{\ell i}'(\underset{\sim}{a},\xi) \triangleq e_\ell'(\underset{\sim}{a},x_i,\xi) \quad , \quad i\in I_\ell \tag{6}$$

$$e_i(\underset{\sim}{a}) \triangleq e(\underset{\sim}{a},x_i) \quad , \quad i\in I_s \tag{7}$$

where it is assumed that a sufficient number of sample points have been chosen so that the discrete approximation problem adequately approximates the continuous problem. I_u, I_ℓ and I_s are appropriate index sets.

The artificial margin ξ allows for certain flexibility in formulating the optimization problem.

BACKGROUND THEORY

Consider a system of real nonlinear functions

$$f_i(\underset{\sim}{a},\xi) \triangleq e_{ui}'(\underset{\sim}{a},\xi) \quad , \quad i\in I_u \tag{8}$$

$$f_i(\underset{\sim}{a},\xi) \triangleq - e_{\ell i}'(\underset{\sim}{a},\xi) \quad , \quad i\in I_\ell \tag{9}$$

Bandler and Charalambous [1] proposed the generalized least pth objective function which is valid for both negative and nonnegative f_i for $i\in I \triangleq I_u \cup I_\ell$ and which alleviates the ill-conditioning resulting from the numerical evaluation of $[\pm f_i(\underset{\sim}{a},\xi)]^{\pm p}$ for very large values of p, namely,

$$U(\underset{\sim}{a},\xi) = M(\underset{\sim}{a},\xi)\Big(\sum_{i\epsilon K}\Big[\frac{f_i(\underset{\sim}{a},\xi)}{M(\underset{\sim}{a},\xi)}\Big]^q\Big)^{\frac{1}{q}} \quad \text{for } M(\underset{\sim}{a},\xi) \neq 0 \tag{10}$$

where

$$M(\underset{\sim}{a},\xi) \triangleq \max_{i\epsilon I} f_i(\underset{\sim}{a},\xi) \tag{11}$$

$$q \triangleq \frac{M(\underset{\sim}{a},\xi)}{|M(\underset{\sim}{a},\xi)|} \cdot p \begin{cases} p > 1 & \text{if } M(\underset{\sim}{a},\xi) > 0 \\ p \geq 1 & \text{if } M(\underset{\sim}{a},\xi) < 0 \end{cases} \tag{12}$$

and

$$K \triangleq \begin{cases} J \triangleq \{ i \,|\, f_i(\underset{\sim}{a},\xi) \geq 0,\ i\epsilon I\} & \text{if } M(\underset{\sim}{a},\xi) > 0 \\ I & \text{if } M(\underset{\sim}{a},\xi) < 0 \end{cases} \tag{13}$$

By minimizing the objective function defined by (10) with a large value of p we should obtain results very close to the minimax optimum [4].

If $\xi=0$, $f_i>0$ indicates that a specification or a response constraint is violated, and $f_i<0$ that a specification is exceeded; $f_i=0$ indicates that a specification is met exactly. It is quite possible that some of the f_i are equal to $-\infty$ in which case they are simply ignored by (10). Also the generalized objective does not allow any of the f_i to be $+\infty$. If the $f_i(\underset{\sim}{a},\xi)$ for $i\epsilon I$ are continuous with continuous partial derivatives, the proposed objective function is continuous with continuous partial derivatives. The objective function (10) and partial derivatives still remain continuous even when, for some i's, the f_i are discontinuous or continuous with discontinuous derivatives, simply because those points are ignored.

The ξ, which is constant during optimization, does not affect the location of the minimax optimum ($p\to\infty$). Its important role, however, is evident for a finite value of p. The value of the parameter ξ can be chosen so that the $M(\underset{\sim}{a},\xi)$ of (11) is always positive or negative during optimization. When $M(\underset{\sim}{a},\xi)$ is positive , only sample points which belong to index set J (13) are considered and, therefore, there is a saving in gradient computation. But in this case it may happen that $M(\underset{\sim}{a},\xi) = 0$, when the function (10) is continuous but the derivatives may be discontinuous. On the rare occasions when this situation causes a failure of the gradient minimization algorithm, one can change the value of ξ and restart the optimization process. If the value of $M(\underset{\sim}{a},\xi)$ is chosen to be negative this possible failure is avoided.

THE COMPUTER PROGRAM FMCLP

We will consider first the program written for minimizing the objective function corresponding to a single specified function. A function f_i is chosen to be the absolute value of a single specified weighted error function (7) for all $i \epsilon I_s$. To alleviate the ill-conditioning for very large values of p, a similar scaling as in (11) was proposed [5], namely,

$$U(\underset{\sim}{a}) = M(\underset{\sim}{a}) \left(\sum_{i \epsilon I_s} \left| \frac{e_i(\underset{\sim}{a})}{M(\underset{\sim}{a})} \right|^p \right)^{\frac{1}{p}} \quad \text{for } 1 < p < \infty \tag{14}$$

where $M(\underset{\sim}{a}) \triangleq \max_{i \epsilon I_s} |e_i(\underset{\sim}{a})|$.

A list and a brief description of the 14 subprograms comprizing FMCLP is given below:

FMCLP	Supplies data for the function minimization and coordinates the other subprograms (see Fig. 1)
S	Defines a specified function over an interval
FAPP	Defines an approximating function over an interval and the gradients w.r.t. variable parameters
W	Defines a weighting function over an interval
WERR	The output of this subprogram has a value of the weighted error at a single point x for a particular vector $\underset{\sim}{a}$
NEWSET	Redefines a sample point set such as to include all the extreme points in the summation of the objective function (14). Quadratic interpolation is used to locate the extreme points more precisely
FUNCT	Keeps the values of the weighted error of each sample point in an array, finds the maximum absolute value and computes the objective function (14) and its gradients
GRDCHK	Checks the gradients w.r.t. all variable parameters before the optimization process starts
FMFPC	Minimizes a function using the Fletcher-Powell method
FMNFC	Minimizes a function using the Fletcher method

INPUT	Prints the input data for the optimization process
FINAL	Prints the optimum solution
WRITE1 and WRITE2	Print the intermediate results, if desired.

S, W and WERR are function subprograms and the others are subroutine subprograms. A user of FMCLP writes S, FAPP and W.

The program terminates when stopping criteria for the Fletcher-Powell or Fletcher method are satisfied or when the relative change in the objective function in two successive iterations is less than a small prescribed quantity.

If the requirement is a minimax approximation it is suitable to sample points in the neighbourhood of the maxima of the weighted error function. As one usually cannot know the positions of the maxima in advance, it is common to space the sample points uniformly. Retaining the maxima and removing from the objective function those sample points which do not substantially contribute to the summation may save computation time. Even more can be done if approximations to the actual maxima replace the sample points in their neighbourhood.

It is assumed that, in the neighbourhood of an extremum, the function is adequately represented by a quadratic form. The function is evaluated at three points, a quadratic interpolation polynomial is fitted to it, and the maximum of this interpolant is obtained. This point replaces one of the initial points.

Selection of the extreme points is significant especially within the first iteration. Once they are found, they do not usually move too far away in the next iterations.

THE COMPUTER PROGRAM FMLPO

Here, we will consider a program written for generalized least pth approximation described earlier. A list and a brief description of the 15 subprograms comprizing FMLPO is given below:

FMLPO	Supplies data for the optimization process and coordinates the other subprograms (see Fig. 2)
FUNCS	Defines upper and lower specified functions
FCTAPP	Defines an approximating function and its gradients w.r.t. variable parameters
W	Defines upper and lower weighting functions

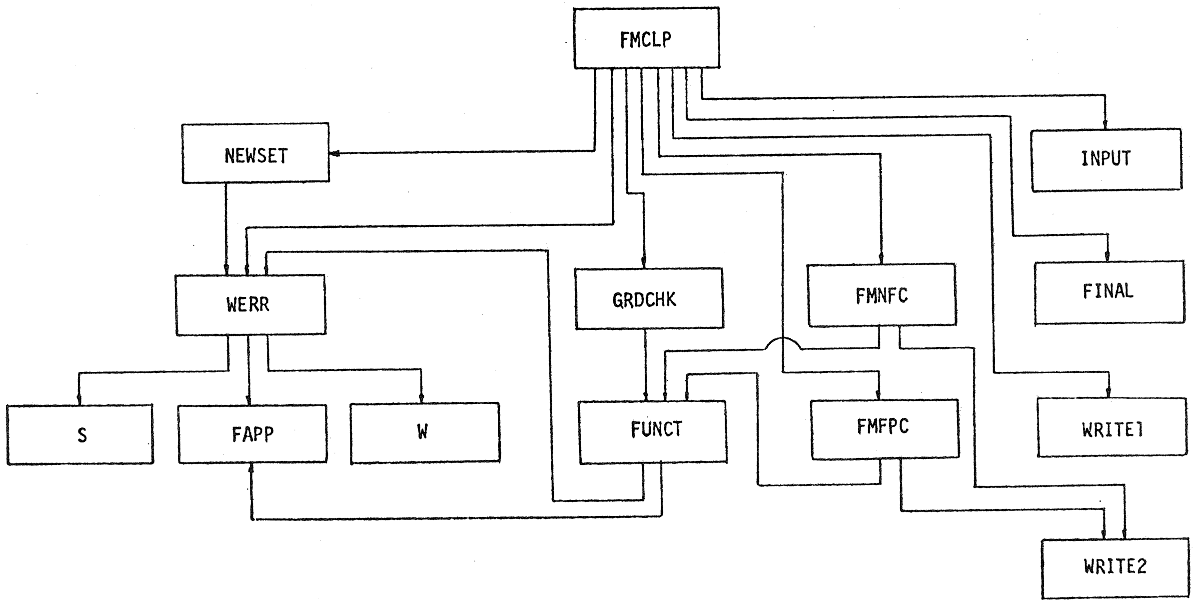

Fig. 1. The organization of FMCLP

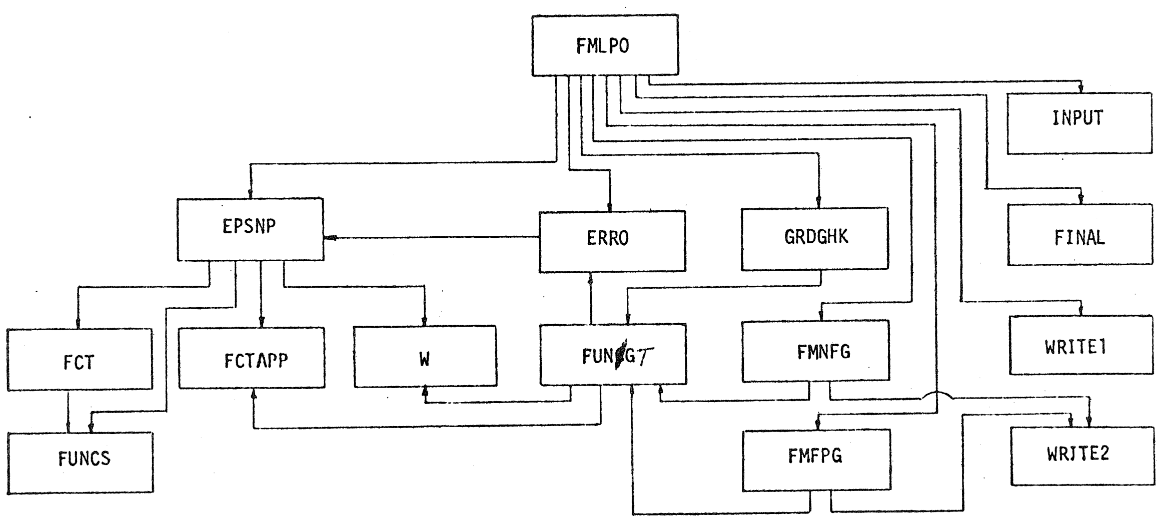

Fig. 2. The organization of FMLPO

TABLE I RESULTS FOR EXAMPLE 1

	$n=10,\ n_s=5$	$n=25,\ n_s=2$	$n=50,\ n_s=1$
$p=2$	$a_1=1.27339$ $a_2=6.54190\times10^{-1}$ $a_3=2.17787\times10^{-1}$ $M(\underset{\sim}{a})=2.05859\times10^{-2}$ $t_M=2.51460\times10^{-1}$ $U(\underset{\sim}{a})=4.71528\times10^{-3}$ f.e.=41 q.i.=2 1.6 sec	$a_1=1.05489$ $a_2=-7.67814\times10^{-1}$ $a_3=1.618192\times10^{-1}$ $M(\underset{\sim}{a})=1.32708\times10^{-2}$ $t_M=2.46234\times10^{-1}$ $U(\underset{\sim}{a})=1.21662\times10^{-2}$ f.e.=30 q.i.=3 2.3 sec	$a_1=1.01687$ $a_2=7.8915\times10^{-1}$ $a_3=1.61435\times10^{-1}$ $M(\underset{\sim}{a})=1.28696\times10^{-2}$ $t_M=2.04081\times10^{-1}$ $U(\underset{\sim}{a})=2.06679\times10^{-2}$ f.e.=36 q.i.=0 5.1 sec
$p=10$	$a_1=7.37873\times10^{-1}$ $a_2=9.2622\times10^{-1}$ $a_3=1.2862\times10^{-1}$ $M(\underset{\sim}{a})=8.9565\times10^{-3}$ $t_M=1.67727\times10^{-1}$ $U(\underset{\sim}{a})=8.4364\times10^{-3}$ f.e.=38 q.i.=3 1.5 sec	$a_1=7.46289\times10^{-1}$ $a_2=-9.23825\times10^{-1}$ $a_3=1.27596\times10^{-1}$ $M(\underset{\sim}{a})=8.76150\times10^{-3}$ $t_M=1.73062\times10^{-1}$ $U(\underset{\sim}{a})=8.2421\times10^{-3}$ f.e.=32 q.i.=3 2.5 sec	$a_1=7.43325\times10^{-1}$ $a_2=9.29377\times10^{-1}$ $a_3=1.2812\times10^{-1}$ $M(\underset{\sim}{a})=8.5446\times10^{-3}$ $t_M=2.04081\times10^{-1}$ $U(\underset{\sim}{a})=9.1834\times10^{-3}$ f.e.=31 q.i.=0 4.5 sec
$p=10^2$	$a_1=6.79369\times10^{-1}$ $a_2=9.55429\times10^{-1}$ $a_3=1.21997\times10^{-1}$ $M(\underset{\sim}{a})=8.9563\times10^{-3}$ $t_M=1.67727\times10^{-1}$ $U(\underset{\sim}{a})=8.2055\times10^{-3}$ f.e.=35 q.i.=1 1.3 sec	$a_1=6.8047\times10^{-1}$ $a_2=-9.5471\times10^{-1}$ $a_3=1.2206\times10^{-1}$ $M(\underset{\sim}{a})=8.7300\times10^{-3}$ $t_M=1.73062\times10^{-1}$ $U(\underset{\sim}{a})=8.1866\times10^{-3}$ f.e.=35 q.i.=1 2.5 sec	$a_1=6.88905\times10^{-1}$ $a_2=9.52106\times10^{-1}$ $a_3=1.2334\times10^{-1}$ $M(\underset{\sim}{a})=7.9450\times10^{-3}$ $t_M=2.04081\times10^{-1}$ $U(\underset{\sim}{a})=8.0045\times10^{-3}$ f.e.=35 q.i.=0 5. sec
$p=10^3$	$a_1=6.7370\times10^{-1}$ $a_2=9.5590\times10^{-1}$ $a_3=1.2168\times10^{-1}$ $M(\underset{\sim}{a})=8.1125\times10^{-3}$ $t_M=1.67727\times10^{-1}$ $U(\underset{\sim}{a})=8.1182\times10^{-3}$ f.e.=28 q.i.=0 1. sec	$a_1=6.77142\times10^{-1}$ $a_2=-9.5556\times10^{-1}$ $a_3=1.21735\times10^{-1}$ $M(\underset{\sim}{a})=8.0905\times10^{-3}$ $t_M=1.73062\times10^{-1}$ $U(\underset{\sim}{a})=8.0957\times10^{-3}$ f.e.=29 q.i.=0 2.3 sec	$a_1=6.8510\times10^{-1}$ $a_2=9.5289\times10^{-1}$ $a_3=1.2294\times10^{-1}$ $M(\underset{\sim}{a})=7.9009\times10^{-3}$ $t_M=2.04082\times10^{-1}$ $U(\underset{\sim}{a})=7.9068\times10^{-3}$ f.e.=26 q.i.=0 4. sec
Total	f.e.=142 for 5.4 sec	f.e.=126 for 9.6 sec	f.e.=128 for 18.6 sec

f.e. denotes number of function evaluations
q.i. denotes number of quadratic interpolations

TABLE II RESULTS FOR EXAMPLE 2

Fletcher method	unconstrained problem		constrained problem	
$p=10^3$	$\xi=0$	$\xi=2.337\times10^{-2}$	$\xi=0$	$\xi=2.337\times10^{-2}$
a_1	3.1525	3.1508	2.9429	2.9422
a_2	4.4203×10^{-1}	4.4165×10^{-1}	4.1069×10^{-1}	4.1070×10^{-1}
a_3	4.4212	4.4194	4.0	3.9998
a_4	4.4159×10^{-1}	4.4169×10^{-1}	4.1069×10^{-1}	4.1070×10^{-1}
a_5	3.1526	3.1508	2.9429	2.9422
M	3.9466×10^{-5}	-2.3330×10^{-2}	4.7403×10^{-5}	-2.3322×10^{-2}
x_M	3.0700×10^{-1}	3.0	8.02×10^{-1}	3.0
U	3.9466×10^{-5}	-2.3330×10^{-2}	4.7403×10^{-5}	-2.3322×10^{-2}
Function evaluations	177	79	55	157
Execution time in seconds	17	13	31	55