

OSA

DESIGN CENTERING USING TOLCAD:

PROGRESS REPORT AND LISTINGS

OSA-86-TC-4-R

June 27, 1986

Optimization Systems Associates

163 Watson's Lane, Dundas, Ontario, Canada

©1983 J. W. Bandler, President

DESIGN CENTERING USING TOLCAD:

PROGRESS REPORT AND LISTINGS

OSA-86-TC-4-R

June 27, 1986

ABSTRACT

This report describes an introduction of the fixed tolerance problem (FTP) option into an existing software package (COMPSHT) for computerized optimization of microwave passive and active circuits. Some possible formulations of the FTP, all using an expedient least-squares objective function within the restrictions of COMPSHT, are suggested. For some of the formulations of the tolerance problem quadratic modeling of the overall least squares objective function is available. Different formulations of the FTP are integrated into the existing package with interactive features allowing the user to define his problem.

A 6-element LC impedance matching circuit is used to illustrate the validity of the approach. TOLCAD is restricted in its utility at this time: some warnings are printed out during execution. They and others are described in an appropriate section of this report.

Results obtained from a special program verify TOLCAD's execution. Further, it was possible to modify the special program to solve tolerance problems using formulations that would require extensive recoding in TOLCAD. Results from the modified program can, therefore, be used to decide on the desirability of further development of TOLCAD itself.

INDEX

Section	No. of Pages
ABSTRACT	1
INTRODUCTION	2
FIXED TOLERANCE PROBLEM: AN OPTION OF TOLCAD	5
GRADIENT CALCULATIONS IN TOLCAD	3
RESTRICTIONS AND LIMITATIONS OF TOLCAD	2
TEST CIRCUIT	6
RESULTS PRODUCED BY TOLCAD AND POST ANALYSIS	14
RESULTS FROM THE SPECIALIZED TEST PROGRAM	31
DISCUSSION AND SUGGESTIONS	2
APPENDIX A	10
APPENDIX B	24
LISTING	38

INTRODUCTION

The purpose of the project described in this report was to introduce the fixed tolerance problem (FTP) into an existing software package (COMPSHT) for computerized optimization of microwave passive and active circuits. The package under consideration, namely TOLCAD, performs the following main functions: (1) circuit analysis, (2) sensitivity analysis, (3) optimization (conventional nominal design), (4) sweep option, (5) analyses for various source and load elements, (6) variable analysis, (7) Monte Carlo analysis, (13) analysis and optimization and (44) random grid search. As a result of this work a new option has been added, namely (8), the fixed tolerance problem.

It is impractical to manufacture circuits using precise parameter values obtained from the process of nominal design. Circuit parameter values are subject to random fluctuations from known nominal values. Therefore, the performance of a manufactured circuit may differ from the one predicted by nominal design and quite often a performance specification violation may result.

If a procedure known as design centering is performed it may lead to yield improvement in the presence of tolerances by moving the original nominal design to a point where it is "better centered" w.r.t. constraints resulting from design specifications.

It is assumed that the fixed tolerance problem (FTP) will usually be solved after an acceptable nominal design has been achieved by traditional CAD. Therefore, all the information contained in the output (results) data file should be accessible to the package when solving the FTP.

The TOLCAD package solves the nominal design problem via a least squares objective function of a fixed format. It was determined

that additional software created to solve the FTP should be minimally intrusive to the existing software (COMPSHT) and should utilize the existing subroutines, variables and arrays already available. Therefore, formulations for the FTP in TOLCAD have been developed to make use of a sum of the sum of squares type of objective function.

The size of the existing package, its nonuniform coding and the limited information provided on how the program works (comments in the source code) dictated the need to create a testing program independent of TOLCAD, basically simulating TOLCAD for specific problems, to verify all results obtained using TOLCAD in the tolerance problem mode of operation. Results obtained from the special program verify TOLCAD's execution. Further, it was possible to modify the special program to solve tolerance problems using formulations that would require extensive recoding in TOLCAD. Results from the modified program can, therefore, be used to decide on the desirability of further development of TOLCAD itself.

FIXED TOLERANCE PROBLEM: AN OPTION OF TOLCAD

After an acceptable nominal design has been achieved the user can ask for the tolerance problem to be solved by selecting option TOL(8) and using the data file generated by the package at the solution to the nominal design problem.

The complete set of functions that TOLCAD can perform is as follows:

AN(1) - analysis

SEN(2) - sensitivity analysis

OPT(3) - optimization (conventional nominal design)

SW(4) - sweep

MAP(5) - analysis for various source and load elements

VAR(6) - variable analysis

MC(7) - Monte Carlo analysis

TOL(8) - fixed tolerance problem (NEW FEATURE)

ANOPT(13) - analysis and optimization

RND(44) - random grid search.

We consider the problem of designing a circuit from components whose values are known only to certain tolerances. Let x^0 denote the nominal value of the parameter x and t the relative tolerance, so that the actual value of the parameter x lies in a known tolerance region which, for one variable, is the interval $[x^0-tx^0,\ x^0+tx^0]$. The extreme points of that interval are called vertices. In general, for an n-parameter design, the tolerance region has 2^n vertices. A solution to the FTP should give a nominal point for which the yield, defined as

$$Y = \frac{\text{number of outcomes which meet specifications}}{\text{total number of outcomes}},$$

is greater than the yield calculated at the solution to the nominal design problem.

When the user declares that he wants to solve the FTP the following options are available:

(1) fixed tolerance problem with all vertices of the tolerance region (Subroutine FTPAQ);

(2) fixed tolerance problem with specified number of worst vertices to be considered (Subroutine FTPBQ);

(3) fixed tolerance problem with a threshold value of the objective function characterizing a given vertex (Subroutine FTPCQ);

(4) fixed tolerance problem with weighted vertices of the tolerance region plus weighted nominal point (Subroutine FTPDQ).

For the first of these options for the tolerance problem quadratic approximation to the least squares function characterizing a given outcome is available.

Subroutine FTPAQ

The subroutine FTPAQ is designed to solve the FTP using all vertices of the tolerance region defined by the user by means of the array of relative tolerances TOL(I), I=1,...,NOPT, where NOPT is the number of toleranced optimization parameters.

The problem is formulated using a least squares objective function defined as

$$EF_{tol} = \sum_{v=1}^{NVERT} EF_v , \qquad (1)$$

where

EF_{tol} is the overall objective function

NVERT is the number of vertices of the tolerance region

EF_v is the function corresponding to vertex v.

The summation is performed outside Subroutine ANALYZ, which is called NVERT times from the subroutine FTPAQ to evaluate EF_{tol}. With the print option "IR" or "OR" the error function generated by the subroutine ANALYZ is of the form (see [1] p.9 and p.23)

$$EF = \frac{1}{NF} \sum_{FI}^{FF} W_1 \mid S_{ii} \mid^2 + W_2 \mid VSWR_i \mid^2 +$$

$$+ W_3 \mid RLG_A - RLG_D \mid^2 + W_5 \mid PAD \mid^2,$$

where

NF is the number of sample frequency points

FI is the first sample frequency

FF is the last sample frequency

i is 1 for the "IR" and 2 for the "OR" option

RLG_A is the actual dB reflection gain

RLG_D is the desired dB reflection gain

PAD is the phase angle difference of S_{ii} at FI and FF.

Subroutine FTPBQ

The subroutine FTPBQ is designed to solve the FTP using an apriori specified number of worst vertices selected from all vertices at the point corresponding to the optimal nominal design. The selection is based on the value of EF_v characterizing a given vertex (or circuit). At the starting point Subroutine ANALYZ is called from the subroutine FTPBQ for all vertices of the tolerance region and the error function values corresponding to all vertices are stored in the

array FERR(I), I=1,...,NVERT. Then all components of the vector FERR(I) are ordered in descending order (according to their value) and vector NRF(I), I=1,...,NVERT, contains the original indices (vertex numbers) of the ordered functions. A limited number (or all) of the worst vertices (NRANK is the number decided by the user) is used in the subsequent iterations of optimization.

The objective function for the problem is defined as

$$EF_{tol} = \sum_{I=1}^{NRANK} FERR(I). \qquad (2)$$

The summation is performed for the top NRANK functions characterizing the worst vertices (worst circuits) of the tolerance region. The initial selection of vertices remains constant throughout the optimization.

Subroutine FTPCQ

The subroutine FTPCQ is designed to solve the FTP using a threshold value of the function EF_v characterizing a given vertex. Only functions with values greater than the threshold contribute to the overall objective function. The user decides on the value of the threshold by specifying the variable TRES. The objective function for the problem is defined as

$$EF_{tol} = \sum_{v=1}^{NVERT} (EF_v - TRES), \qquad (3)$$

where

$$EF_v - TRES = \begin{cases} 0 \text{ if } EF_v \leq TRES \\ EF_v - TRES \text{ if } EF_v \geq TRES. \end{cases} \qquad (4)$$

Subroutine FTPDQ

The subroutine FTPDQ is designed to solve the FTP using all vertices of the tolerance region and the nominal point. The functions characterizing the vertices and the nominal point are weighted appropriately to emphasize the relative contribution of the vertices and the nominal point to the overall objective function of the FTP.

The objective function for the FTP is defined as

$$EF_{tol} = \sum_{v=1}^{NVERT+1} W_v * EF_v, \qquad (5)$$

where v=NVERT+1 denotes the nominal point. It is felt that the nominal point should be weighted more heavily than the vertices since the solution to the FTP should not drift far away from the solution to the solution to the nominal design problem. In the test problem under consideration the weighting factor $W_{v=NVERT+1}$=NVERT was used. For v = 1 to NVERT W_v = 1.

Reference

[1] COMPACT, Version 5.1, User's Manual.

GRADIENT CALCULATIONS IN TOLCAD

Gradients w.r.t. optimization parameters are calculated in Subroutine ANALYZ by perturbations using a relative perturbation factor a. The subroutine SEARCH uses transformed optimization variables, so the gradients supplied by ANALYZ should also be calculated w.r.t. transformed variables. This section explains how ANALYZ supplies gradients w.r.t. transformed variables using function values calculated based on original variables. This discussion also applies to the fixed tolerance problem and the gradients involved in solving it. The discussion is presented for one variable for simplicity. The extension to n variables is obvious.

Let x denote the original (untransformed) parameter of the nominal design. The numerical value of that parameter is specified in the input data file. Let

$$y = \log x$$

denote the transformed variable that is used by the subroutine SEARCH. Of course the reverse transformation gives

$$x = e^{y}.$$

The gradient g of function f(x) w.r.t. y is calculated by perturbation as

$$g_y \cong \frac{f(x+\Delta x) - f(x)}{\Delta x}\, x = \frac{f(x+ax) - f(x)}{ax}\, x$$

using

$$\frac{df}{dy} = \frac{df}{dx}\frac{dx}{dy} \quad \text{and} \quad \Delta x = ax\ .$$

Recognizing that

$$\frac{df}{dx} \cong \frac{f(x+ax) - f(x)}{ax}$$

$$\frac{dx}{dy} = e^y \text{ and } x = e^y$$

we obtain

$$g_y \cong \frac{f(x+ax) - f(x)}{a},$$

where a is the relative perturbation.

For the tolerance problem we express x using the relative tolerance t as

$$x = x^0 + tx^0 = x^0(1+t),$$

where x^0 denotes the nominal value of variable x.

The gradient of f(x) w.r.t. x^0 is

$$g_{x^0} \cong \frac{f(x+ax) - f(x)}{ax} \quad (1+t),$$

and the gradient w.r.t. the transformed nominal value y^0 is

$$g_{y^0} \cong \frac{f(x+ax) - f(x)}{ax} \quad (1+t)\ x^0,$$

which, by substituting $x = x^0(1+t)$, reduces to

$$g_{y^0} \cong \frac{f(x+ax) - f(x)}{a}.$$

The conclusion is that for the tolerance problem Subroutine ANALYZ already calculates gradients w.r.t. the transformed nominal point y^0 as required by the algorithm.

RESTRICTIONS AND LIMITATIONS OF TOLCAD

The purpose of the project described in this report was to investigate the feasibility of using a least squares type of objective function for the fixed tolerance problem. Therefore, the software developed and integrated with the existing package for computerized optimization of microwave circuits is experimental in nature. This results in certain restrictions and limitations of which the user should be aware in order to avoid problems when using TOLCAD. The following restrictions apply when this version of TOLCAD is run in the tolerance mode of operation (option 8).

Limitation on Number of Variables

The number of variables used for the tolerance problem is limited to 6. Problems with more than 6 variables can be solved after modifying dimensions of several arrays in TOLCAD.

All Parameters must be Toleranced

It is assumed that all parameters specified in the input data file are optimization variables and that all parameters are toleranced. If certain parameters are not to be toleranced the user should enter zero tolerances for those parameters.

Restriction on Quadratic Approximation

The quadratic approximation option available in TOLCAD may be used for all versions of the FTP, however, at this time the use of the quadratic model is implemented only for option 1 of the FTP (Subroutine FTPAQ).

Option 4 Handles Exactly 6 Variables

Option 4 of the FTP (Subroutine FTPDQ) at this time can handle problems with exactly 6 variables since array which handles the weighting factors for vertices and the nominal point is hard coded to

reflect the test problem under consideration. In a future version of TOLCAD weighting factors could be entered by the user interactively.

TEST CIRCUIT

A 6-element LC matching circuit (Fig. 1) to match a 1 ohm load to a 3 ohm generator over the frequency range 0.0796 - 0.1876 MHz has been used as a test problem. All six reactive components are optimization variables and 21 uniformly spaced sample points are used in the frequency band. This relatively large number of sample points is not necessary. Less than half this number should yield adequate results for practical purposes.

The data file shown in Fig. 2 was used as an input data file for the tolerance problem defined for this LC matching circuit considered. In Fig. 3 we show a typical run of TOLCAD using option TOL(8), Version 1.

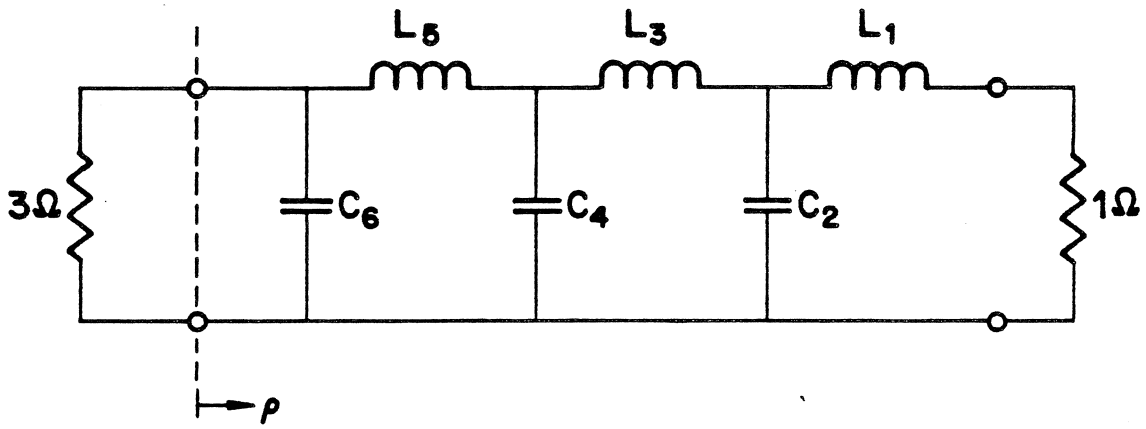


Fig. 1 Six element LC matching circuit used as a test problem for the FTP.

```
* LC MATCHING CIRCUIT EXAMPLE USED AS A TEST PROBLEM
* FOR TOLCAD FEB.24, 1986.
*
CAP AA SE -.32285E+06
IND DD SE -2896.4
CAP BB SE -.7609E+06
IND EE SE -2281.2
CAP CC SE -.96591E+06
IND FF SE -967.59
CON AA T2 1 0
CON BB T2 2 0
CON CC T2 3 0
CON DD T2 1 2
CON EE T2 2 3
CON FF T2 3 4
DEF AA T2 1 4
PRI AA IR 3 1
END
0.0795774 0.1876436 0.00540331
END
0.000001
1 0 0 0
END
```

Fig. 2 Data file for the FTP. Starting values correspond to the conventional optimal nominal design with zero tolerances.

```
'FILE NAME' OR 'QUIT'
*INPUT*  RUNTST
AN(1),SEN(2),OPT(3),SW(4),MAP(5),VAR(6),MC(7),TOL(8),ANOPT(13),RND(44)

*INPUT*  8
DO YOU WANT TO USE QUADRATIC APPROXIMATION FOR THE FIXED TOLERANCE
 PROBLEM (1/0)
*INPUT*  0
ENTER TOLERANCES FOR ALL COMPONENTS TOL(1)...TOL(NOPT)
*INPUT*  6*0.05
SELECT VERSION OF THE FTP:
 1 - FTP WITH ALL VERTICES OF THE TOLERANCE REGION
 2 - FTP WITH SPECIFIED NUMBER OF WORST VERTICES
 3 - FTP WITH A THRESHOLD VALUE OF THE OBJECTIVE FUNCTION
 4 - FTP WITH WEIGHTED VERTICES PLUS WEIGHTED NOMINAL
*INPUT*  1


OPTIMIZATION BEGINS WITH FOLLOWING VARIABLES AND GRADIENTS

     VARIABLES                  GRADIENTS
  ( 1):  .32285E+06          ( 1):  .50970E-01
  ( 2):  2896.4              ( 2):  .50692
  ( 3):  .76090E+06          ( 3):  .81277
  ( 4):  2281.2              ( 4):  .82049
  ( 5):  .96591E+06          ( 5):  .52214
  ( 6):  967.59              ( 6):  .45962E-01
 ERR. F.=       .6854711
        ____****____
HOW MANY ITERATIONS BEFORE NEXT STOP ('0' RESULTS IN FINAL ANALYSIS.)
WANT INTERMEDIATE PRINTS (YES=1,NO=0). TYPE TWO NUMBERS; I,J
*INPUT*  10 1
  ( 1):  .32227E+06          ( 1):  .81235
  ( 2):  2845.2              ( 2):  1.4415
  ( 3):  .73943E+06          ( 3): -1.6397
  ( 4):  2216.2              ( 4): -1.6862
  ( 5):  .94831E+06          ( 5):  1.4016
  ( 6):  966.03              ( 6):  .82774
 ERR. F.=       .6791969
        ____****____
  ( 1):  .31935E+06          ( 1):  .68504
  ( 2):  2758.8              ( 2):  .27091
  ( 3):  .72927E+06          ( 3): -.49467
  ( 4):  2186.1              ( 4): -.53509
  ( 5):  .91937E+06          ( 5):  .23469
  ( 6):  957.30              ( 6):  .69999
 ERR. F.=       .6437066
        ____****____
```

Fig. 3 Printout of a typical run using option 8 of TOLCAD.

```
 ( 1):  .30120E+06          ( 1):  .51310E-01
 ( 2):  2705.3              ( 2):  .26898
 ( 3):  .71056E+06          ( 3): -.45003
 ( 4):  2136.3              ( 4): -.15909
 ( 5):  .90358E+06          ( 5):  .46875
 ( 6):  901.98              ( 6): -.10043
ERR. F.=       .6138338
       ----****----
 ( 1):  .30083E+06          ( 1): -.29929E-01
 ( 2):  2706.9              ( 2):  .36102
 ( 3):  .71128E+06          ( 3): -.30181
 ( 4):  2133.1              ( 4): -.29767
 ( 5):  .90259E+06          ( 5):  .37145
 ( 6):  902.05              ( 6): -.32893E-01
ERR. F.=       .6135072
       ----****----
 ( 1):  .30118E+06          ( 1):  .32241E-02
 ( 2):  2687.2              ( 2):  .24777E-01
 ( 3):  .70786E+06          ( 3):  .28392E-01
 ( 4):  2128.0              ( 4):  .88147E-01
 ( 5):  .89212E+06          ( 5): -.66907E-01
 ( 6):  902.43              ( 6): -.36364E-01
ERR. F.=       .6111938
       ----****----
 ( 1):  .30099E+06          ( 1): -.14274E-01
 ( 2):  2680.5              ( 2): -.41069E-01
 ( 3):  .70836E+06          ( 3):  .74057E-01
 ( 4):  2125.5              ( 4):  .78679E-01
 ( 5):  .89339E+06          ( 5): -.42031E-01
 ( 6):  903.09              ( 6): -.16483E-01
ERR. F.=       .6110591
       ----****----
 ( 1):  .30084E+06          ( 1): -.13567E-01
 ( 2):  2678.8              ( 2): -.20231E-01
 ( 3):  .70782E+06          ( 3):  .46770E-01
 ( 4):  2123.5              ( 4):  .48047E-01
 ( 5):  .89297E+06          ( 5): -.18214E-01
 ( 6):  902.52              ( 6): -.14186E-01
ERR. F.=       .6110020
       ----****----
 ( 1):  .30047E+06          ( 1): -.56177E-02
 ( 2):  2672.7              ( 2):  .56321E-02
 ( 3):  .70598E+06          ( 3):  .12122E-02
 ( 4):  2118.1              ( 4): -.52064E-03
 ( 5):  .89099E+06          ( 5):  .75512E-02
 ( 6):  901.65              ( 6): -.39252E-02
ERR. F.=       .6109298
       ----****----
```

Fig. 3 (continued) Printout of a typical run using option 8 of TOLCAD.

```
 ( 1):  .30052E+06          ( 1): -.67212E-03
 ( 2):  2670.9              ( 2):  .20743E-02
 ( 3):  .70563E+06          ( 3): -.14833E-02
 ( 4):  2116.5              ( 4): -.17600E-02
 ( 5):  .89034E+06          ( 5):  .32658E-02
 ( 6):  901.29              ( 6): -.12378E-02
ERR. F.=      .6109245
       ----****----
 ( 1):  .30052E+06          ( 1): -.67212E-03
 ( 2):  2670.9              ( 2):  .20743E-02
 ( 3):  .70563E+06          ( 3): -.14833E-02
 ( 4):  2116.5              ( 4): -.17600E-02
 ( 5):  .89034E+06          ( 5):  .32658E-02
 ( 6):  901.29              ( 6): -.12378E-02
ERR. F.=      .6109245
       ----****----
GRADIENT TERMINATION WITH ABOVE VALUES. FINAL ANALYSIS FOLLOWS

     .080      .157       88.7      1.37:1    -16.09      2.88       .92
     .085      .120       81.6      1.27:1    -18.44      3.02       .73
     .090      .084       74.5      1.18:1    -21.48      3.10       .51
     .096      .052       67.3      1.11:1    -25.76      3.11       .30
     .101      .022       60.1      1.05:1    -33.12      3.06       .12
     .107      .003     -127.8      1.01:1    -49.46      2.99      -.02
     .112      .024     -134.6      1.05:1    -32.28      2.90      -.10
     .117      .040     -141.9      1.08:1    -27.87      2.81      -.14
     .123      .051     -149.2      1.11:1    -25.77      2.74      -.14
     .128      .057     -156.7      1.12:1    -24.82      2.70      -.12
     .134      .058     -164.2      1.12:1    -24.68      2.68      -.09
     .139      .054     -171.8      1.12:1    -25.28      2.69      -.04
     .144      .046     -179.6      1.10:1    -26.69      2.73      -.00
     .150      .034      172.5      1.07:1    -29.27      2.80       .03
     .155      .020      164.4      1.04:1    -34.07      2.89       .03
     .161      .004      156.5      1.01:1    -48.76      2.98       .01
     .166      .013      -32.6      1.03:1    -38.04      3.06      -.04
     .171      .027      -41.2      1.06:1    -31.40      3.12      -.11
     .177      .037      -50.1      1.08:1    -28.57      3.14      -.18
     .182      .041      -59.4      1.09:1    -27.71      3.12      -.22
     .188      .035      -69.0      1.07:1    -29.00      3.07      -.20
SAVE OPTIMIZED RESULTS (Y/N)
*INPUT*  N

'FILE NAME' OR 'QUIT'
*INPUT*  QUIT
```

Fig. 3 (continued) Printout of a typical run using option 8 of TOLCAD.

RESULTS PRODUCED BY TOLCAD AND POST ANALYSIS

It is important to examine the validity of the options included at this time in TOLCAD for the fixed tolerance problem. We do this by investigating the computed yield versus reflection coefficient specification achievable for each candidate design solution. The yield is calculated after a suitable Monte Carlo analysis. This Monte Carlo analysis and the graphical output are obtained by software separate from TOLCAD.

For the purpose of this comparison a relative tolerance of 5% of the nominal value was assumed for each element of the circuit under consideration. A Monte Carlo analysis is performed for each design using 1000 random outcomes with circuit parameters having actual values within the assumed fixed tolerance range. The random number generator used for this purpose generates uniformly distributed points. A yield is obtained by computing the percentage of the 1000 circuits that satisfy a particular reflection coefficient specification.

For example, in Fig. 1a, representing the zero tolerance nominal solution obtainable by COMPSHT or TOLCAD using option OPT(3), we see that 70.7% of the possible outcomes are capable of satisfying a specification of 0.1535 on their reflection coefficient. A "normalized distribution" as shown in Fig. 1b is obtained by taking the derivative of the yield curve.

A similar analysis can be performed on circuits representing the vertices of the tolerance region, where instead of Monte Carlo generated circuits we use circuits whose parameter values correspond to vertices of the tolerance region. The results of the so-called "vertex analysis" are shown in Fig. 2a and Fig. 2b for the zero

tolerance nominal design. This type of yield and distribution analysis is used as a basis for the comparison of different options of the fixed tolerance problem.

Figs. 3 and 4 show the results of Monte Carlo and vertex analysis, respectively, for the solution to the FTP obtained using all vertices of the tolerance region (option (1), Subroutine FTPAQ). By comparing the distribution of "Monte Carlo outcomes" for the zero tolerance nominal design where the distribution spans the range of reflection coefficient values from 0.1036 to 0.2351, to that of the FTP solution using all vertices where the distribution spans the range from 0.1295 to 0.1848, we conclude that the least-squares objective function approach to the fixed tolerance problem is valid and results in yield improvement for reflection coefficient values greater than 0.167.

Figs. 5 and 6 show the results of Monte Carlo and vertex analyses, respectively, for the solution to the FTP obtained using those 20 vertices of the tolerance region whose least squares function values are largest (option (2), Subroutine FTPBQ). Figs. 7 and 8 show corresponding results for the solution to the FTP obtained using a threshold value for the least squares function characterizing a vertex (option (3), Subroutine FTPCQ).

Figs. 9 and 10 show the results of Monte Carlo and vertex analyses, respectively, for the solution to the FTP obtained using weighting factors of 1.0 for vertices and 64 for the nominal point (option (4), Subroutine FTPDQ).

All plots are accompanied by data files showing the specific numerical values used to produce the plots. These data files are in Appendix A.

It should be very clearly understood that the Monte Carlo and vertex analyses presented in this section are not options of TOLCAD. They were performed using specialized software independent of TOLCAD. It is possible, however, that future versions of TOLCAD could contain such analyses as additional options to the existing ones.

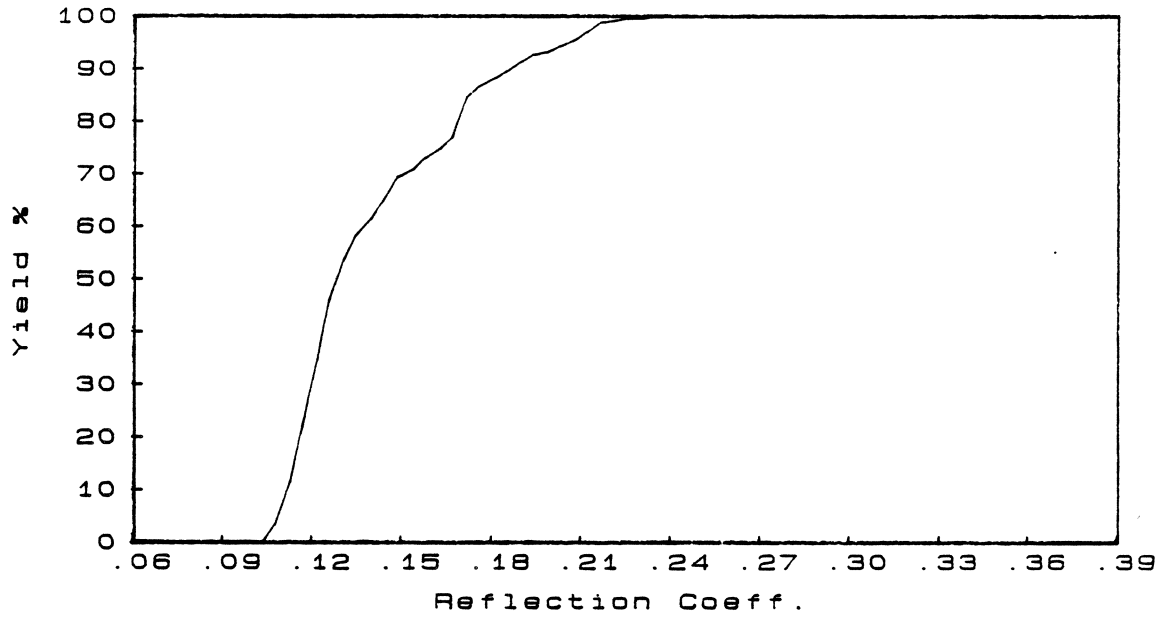
CPU Execution Times

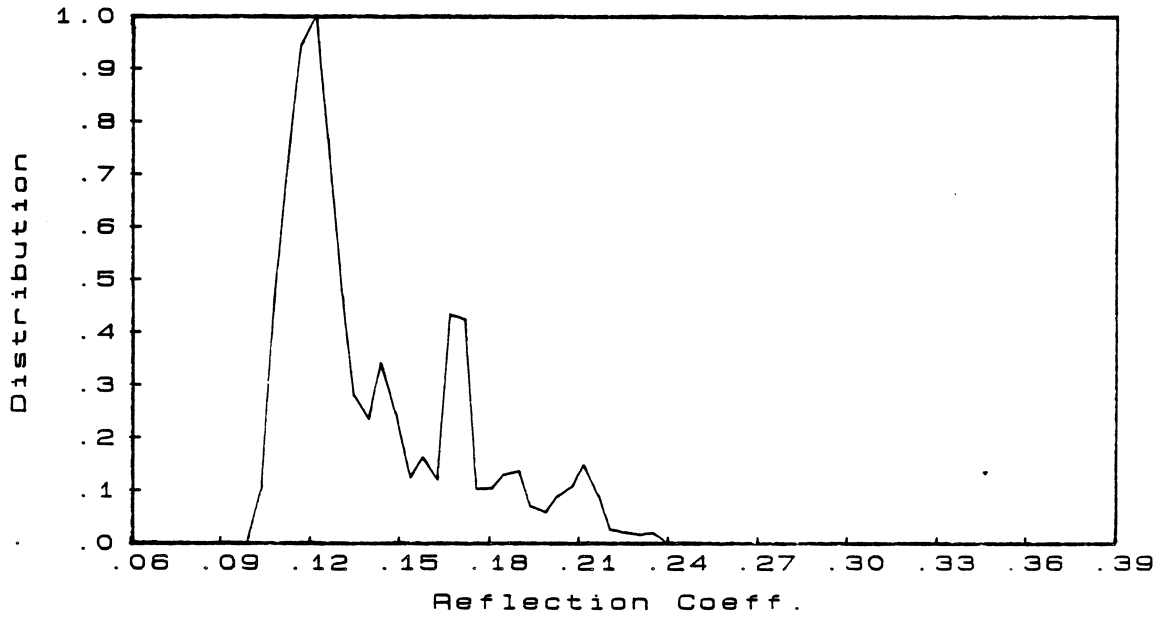
The execution times (in seconds) for different options of the fixed tolerance problem compared with the zero tolerance nominal design problem are as follows.

Zero tolerance nominal design	46
FTP with all vertices (Subroutine FTPAQ)	1500
FTP with top 20 vertices (Subroutine FTPBQ)	815
FTP with a threshold value (Subroutine FTPCQ)	3500
FTP with all vertices plus nominal (Subroutine FTPDQ)	1600

SUMMARY OF RESULTS

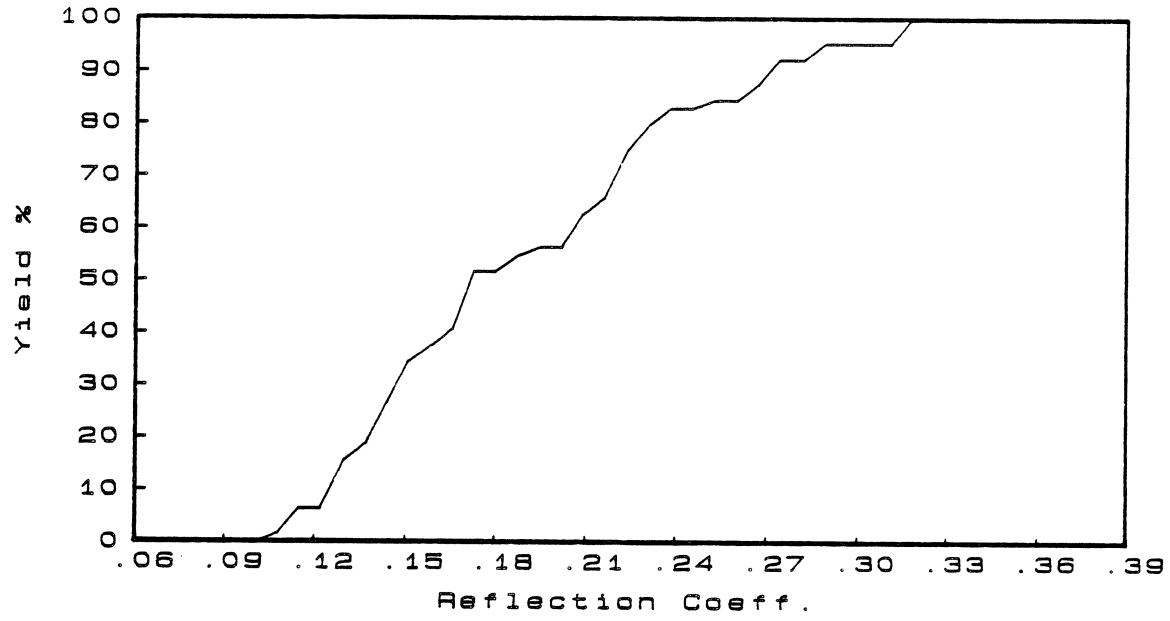
OPTION	SUBROUTINE	MONTE-CARLO DISTRIBUTION	VERTEX DISTRIBUTION
Zero Tolerance Nominal Design		.1036 - .2351	.1078 - .3178
All Vertices	FTPAQ	.1295 - .1848	.1192 - .2450
Top 20 Vertices	FTPBQ	.1891 - .2362	.1763 - .2500
Threshold Value	FTPCQ	.1626 - .2146	.1487 - .2677
Vertices Plus Nominal	FTPDQ	.1107 - .1644	.1205 - .2515

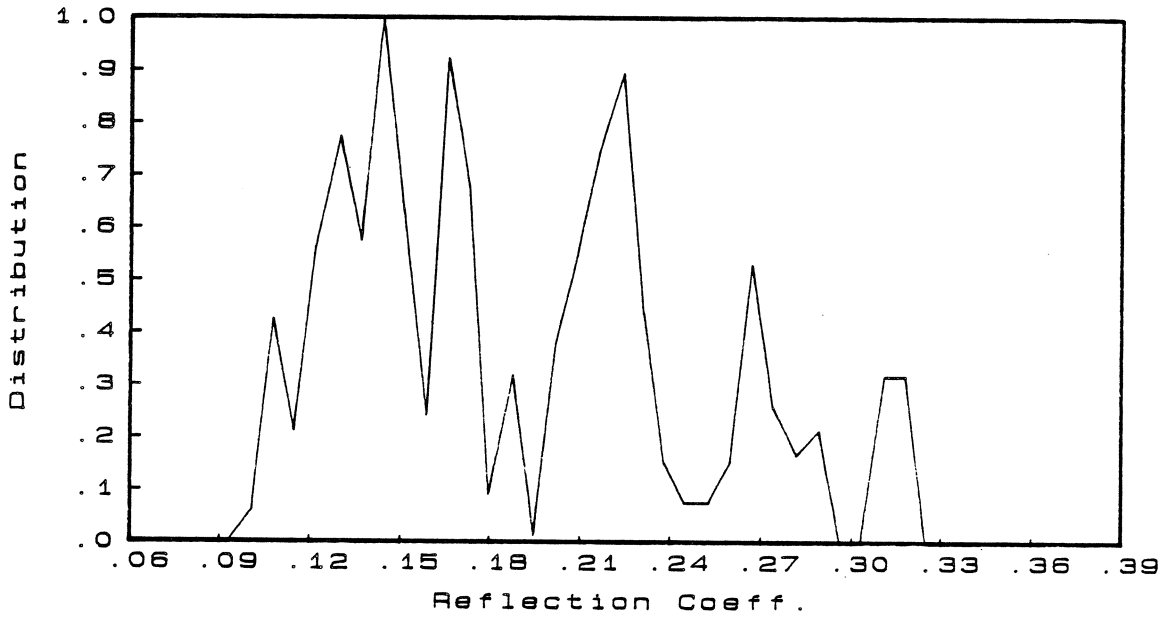




FILE: MCNOM
RANGES OF REFLECTION COEFFICIENTS : .1036 .2351

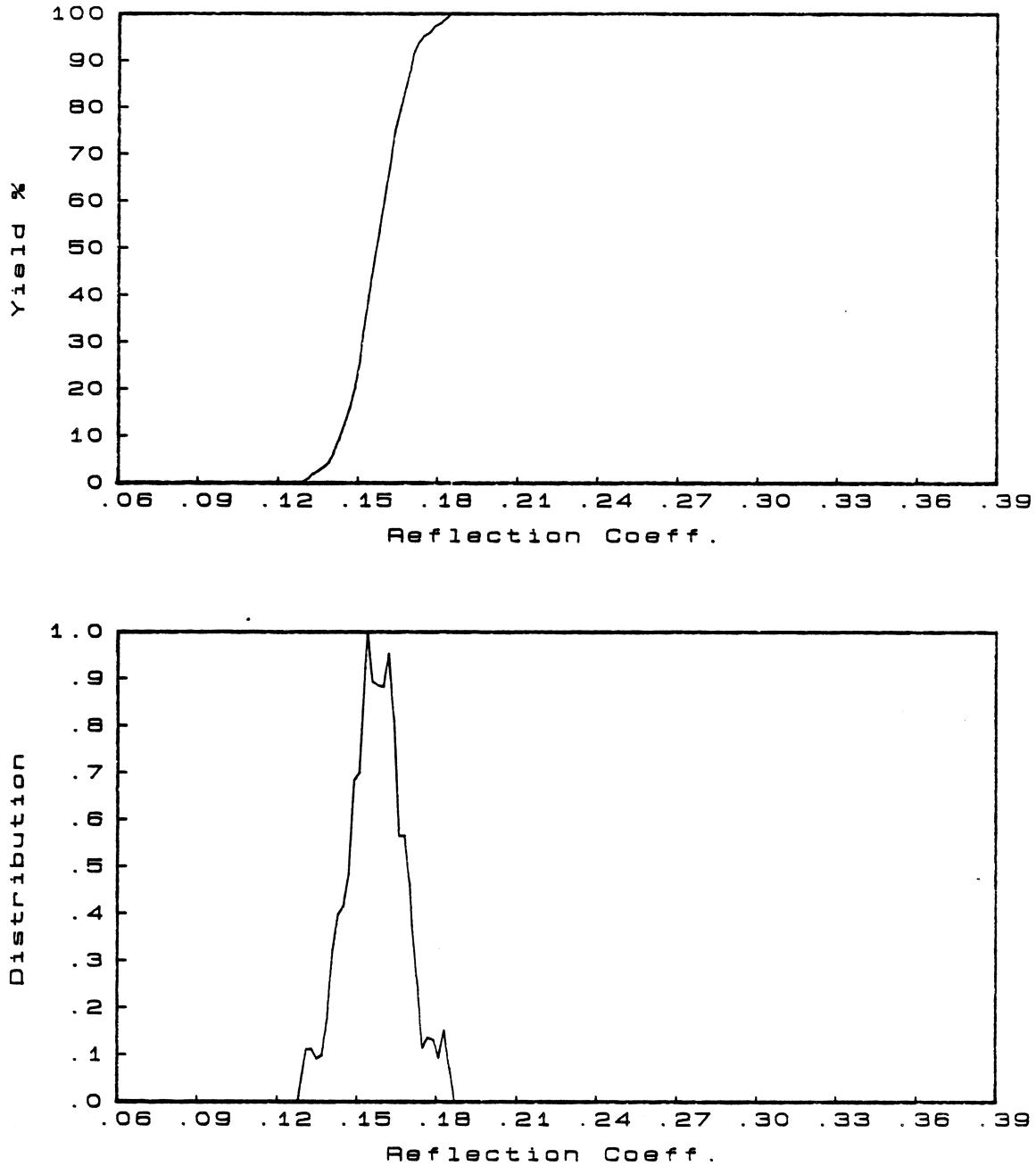
Fig. 1 Monte Carlo analysis for the conventional nominal design obtained using a least squares objective function with zero tolerances.





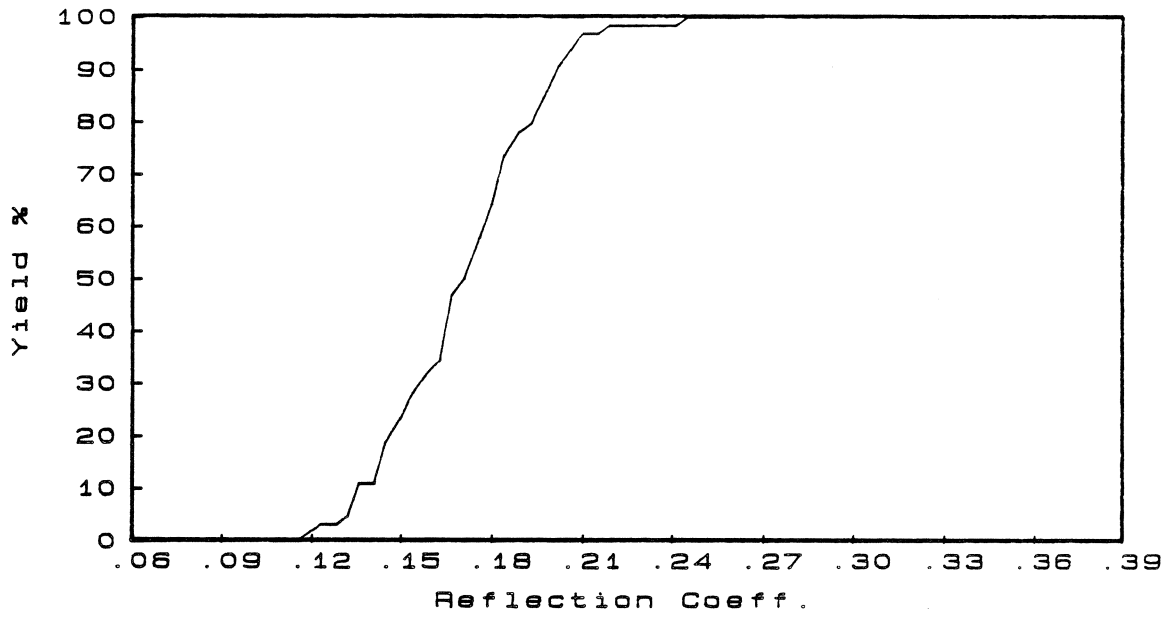
FILE: VANOM
RANGES OF REFLECTION COEFFICIENTS : .1078 .3178

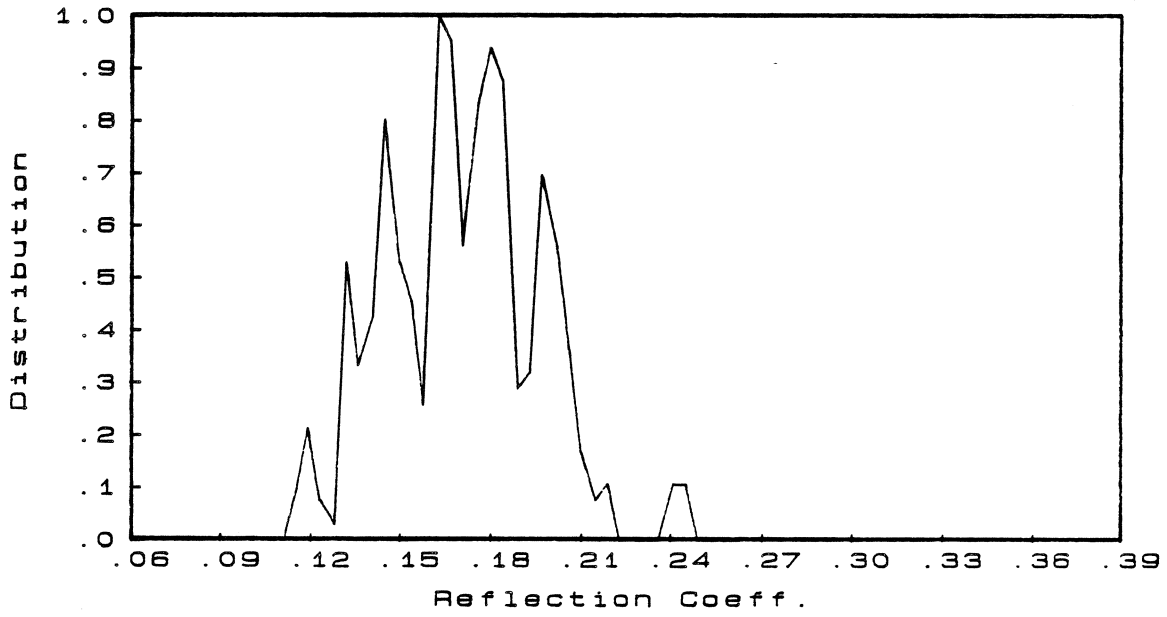
Fig. 2 Vertex analysis for the conventional nominal design obtained using a least squares objective function with zero tolerances.



FILE: MCTOLA
RANGES OF REFLECTION COEFFICIENTS : .1295 .1848

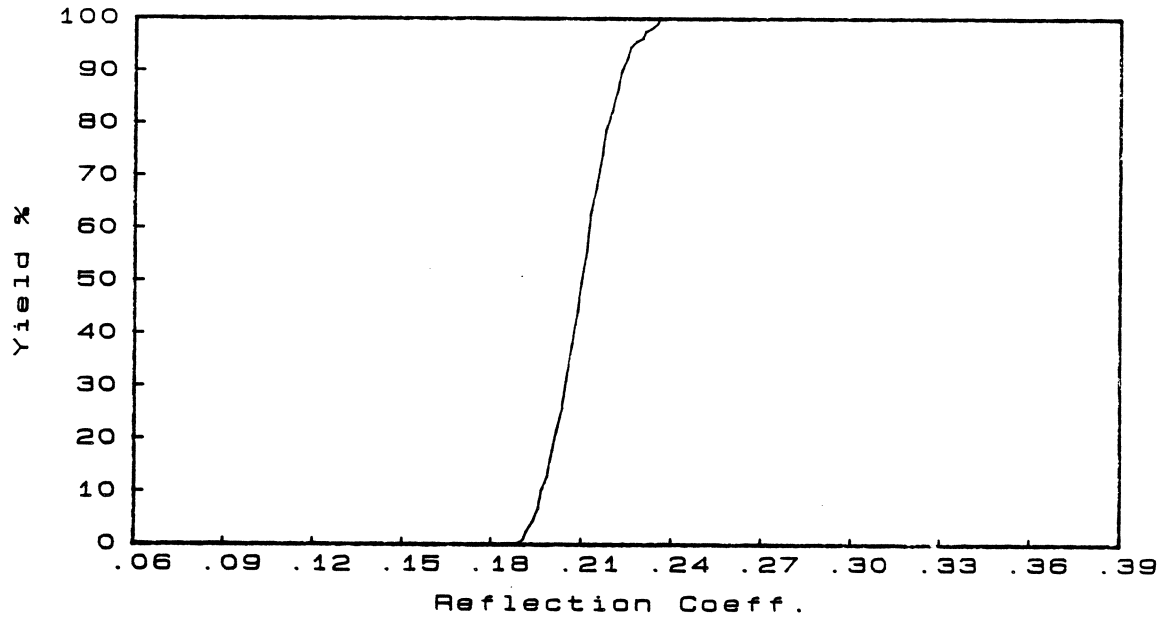
Fig. 3 Monte Carlo analysis of the solution to the FTP obtained using all vertices of the tolerance region.

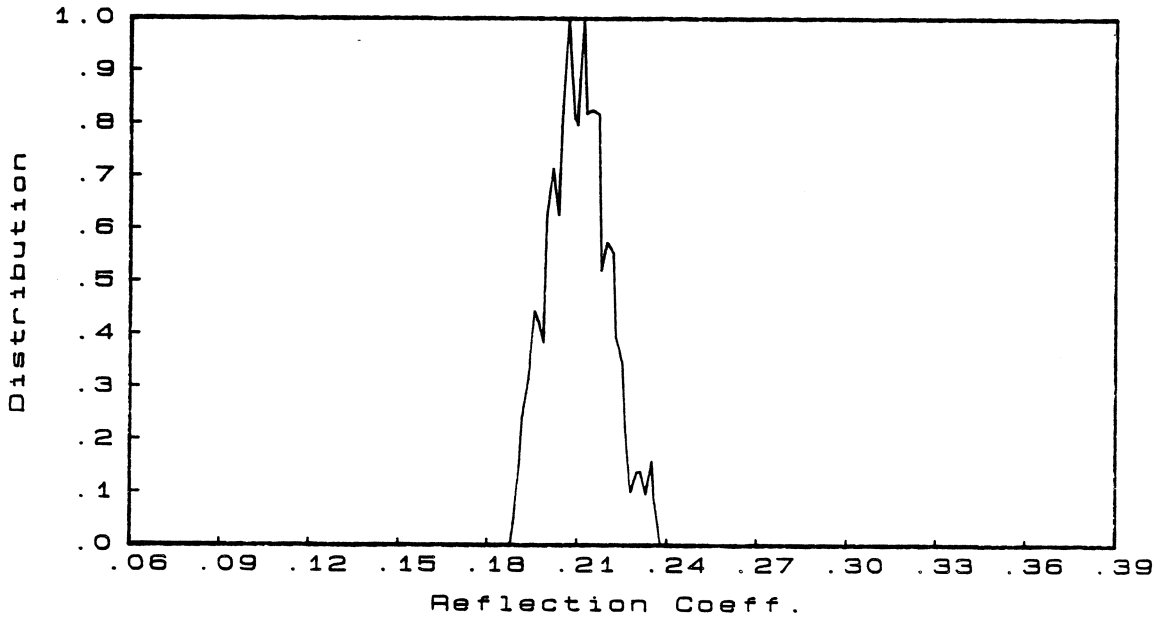




FILE: VATOLA
RANGES OF REFLECTION COEFFICIENTS : .1192 .2450

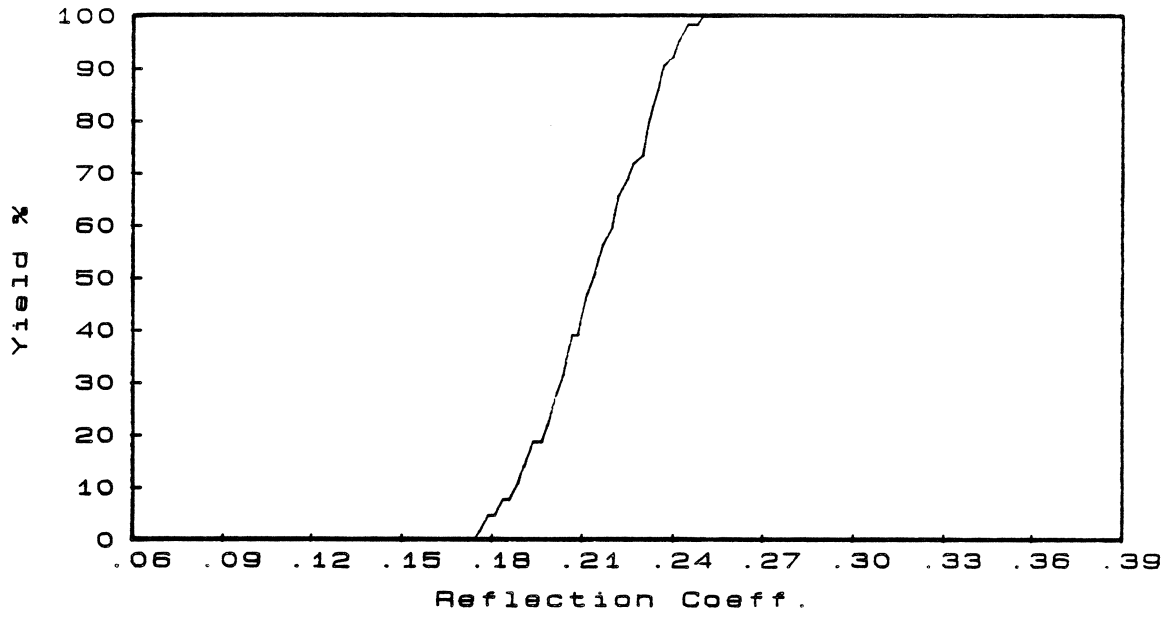
Fig. 4 Vertex analysis of the solution to the FTP obtained using all vertices of the tolerance region.

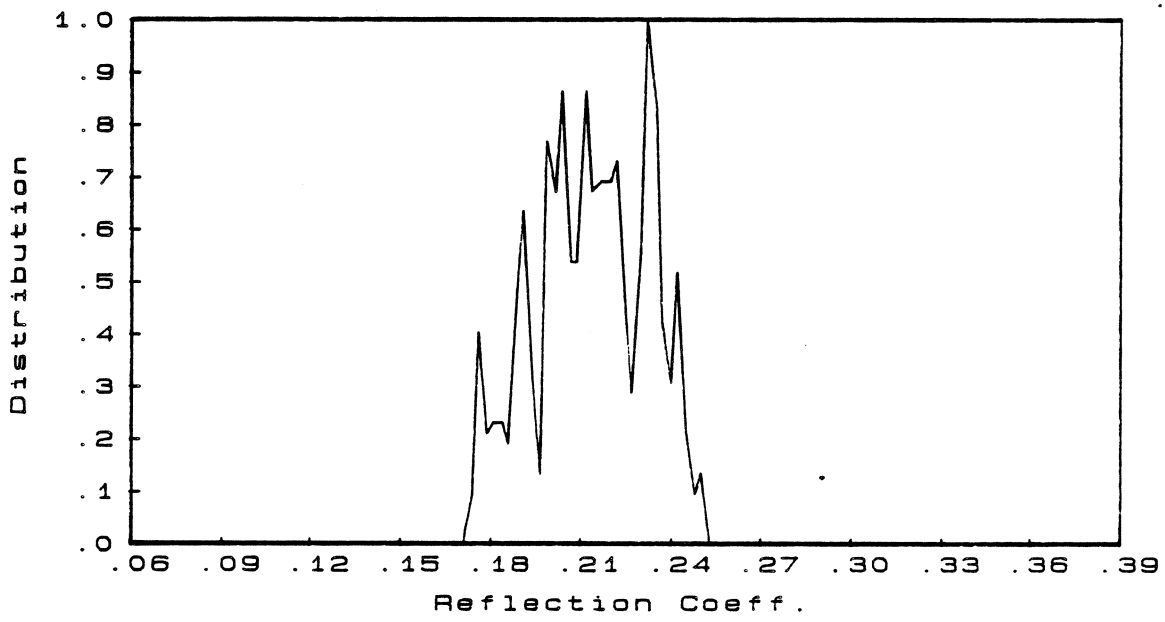




FILE: MCTOLB
RANGES OF REFLECTION COEFFICIENTS : .1891 .2362

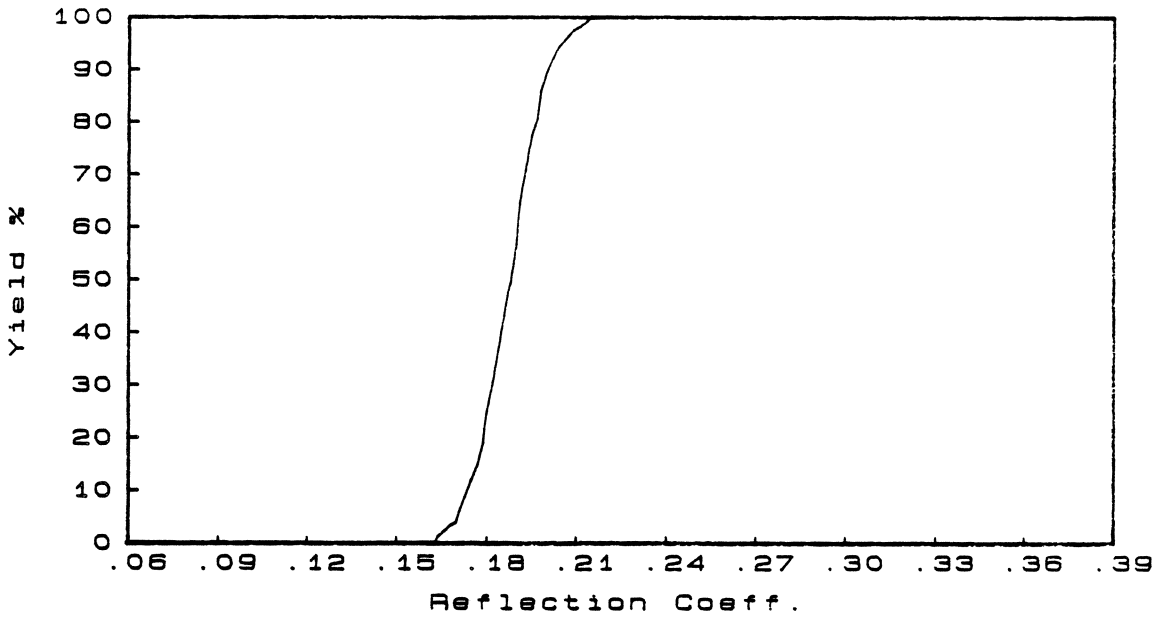
Fig. 5 Monte Carlo analysis of the solution to the FTP obtained using only the 20 worst vertices of the tolerance region.

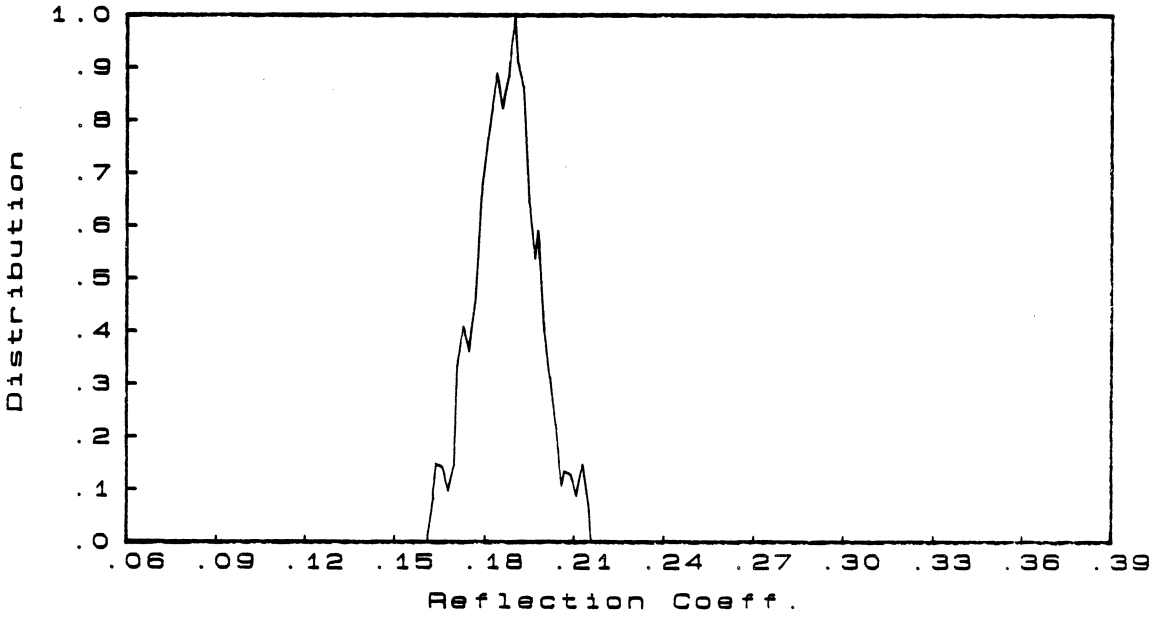




FILE: VATOLB
RANGES OF REFLECTION COEFFICIENTS : .1763 .2500

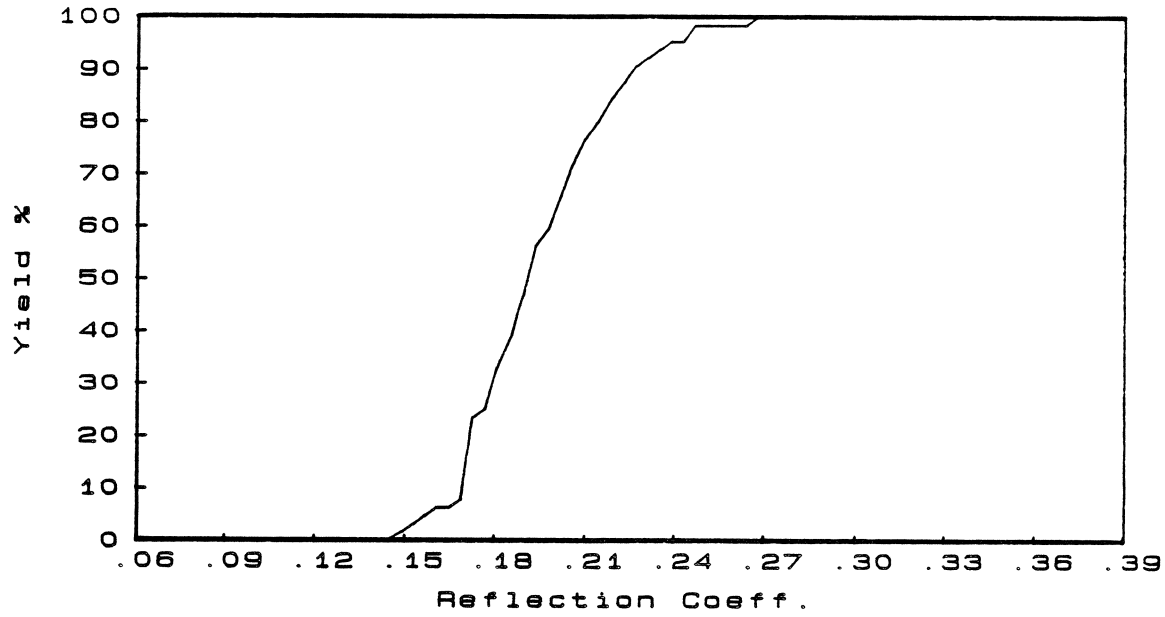
Fig. 6 Vertex analysis of the solution to the FTP obtained using only the 20 worst vertices of the tolerance region.

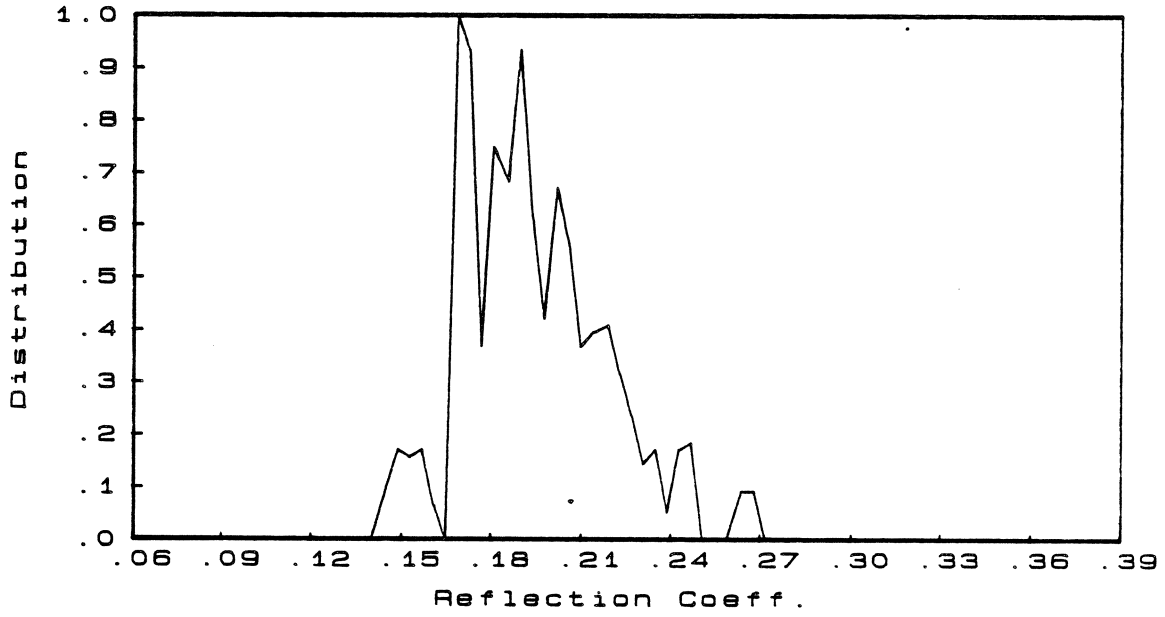




FILE: MC1N64
RANGES OF REFLECTION COEFFICIENTS : .1626 .2146

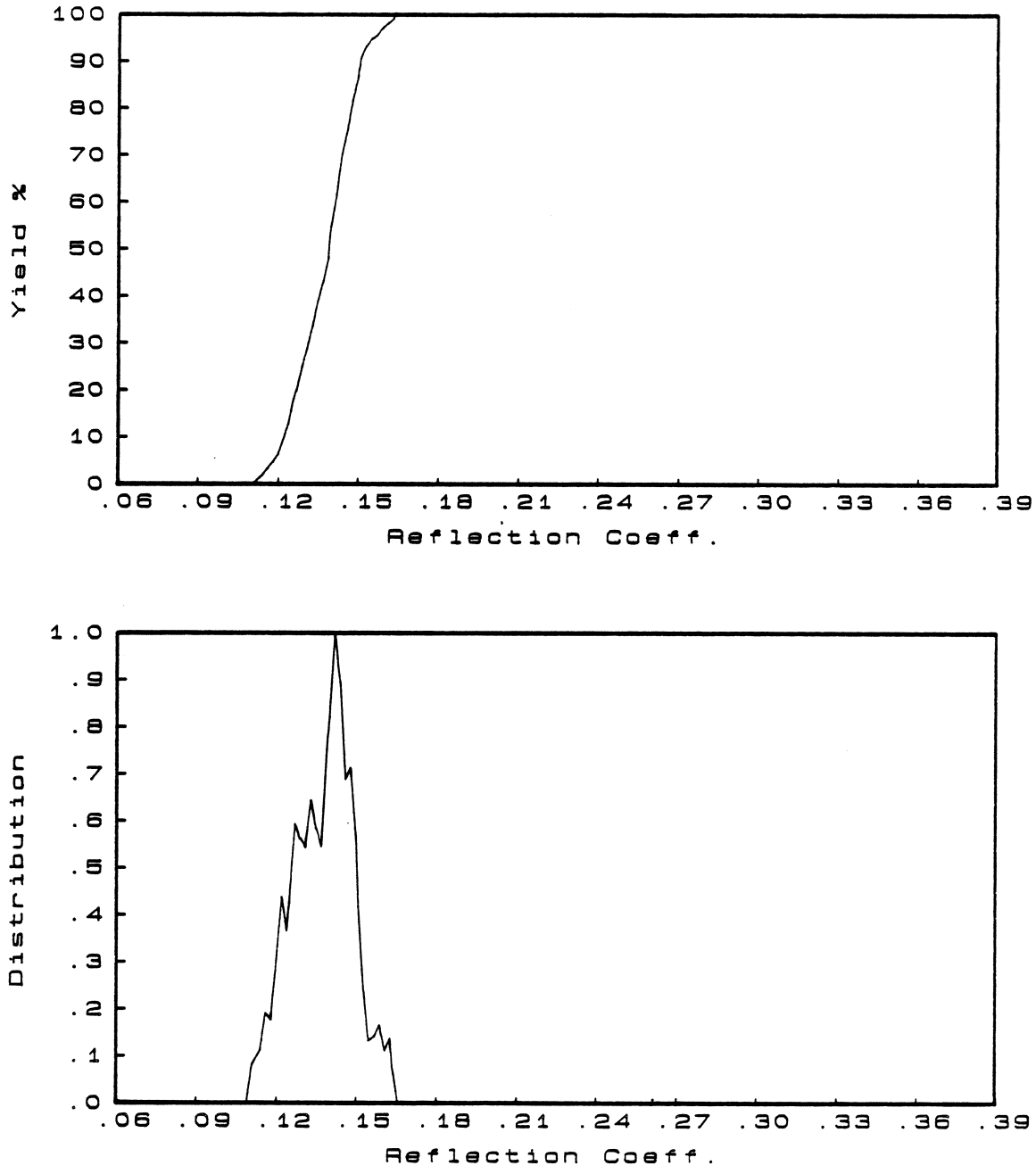
Fig. 7 Monte Carlo analysis of the solution to the FTP obtained using using the threshold value of 0.015 for the least squares function characterizing a vertex.





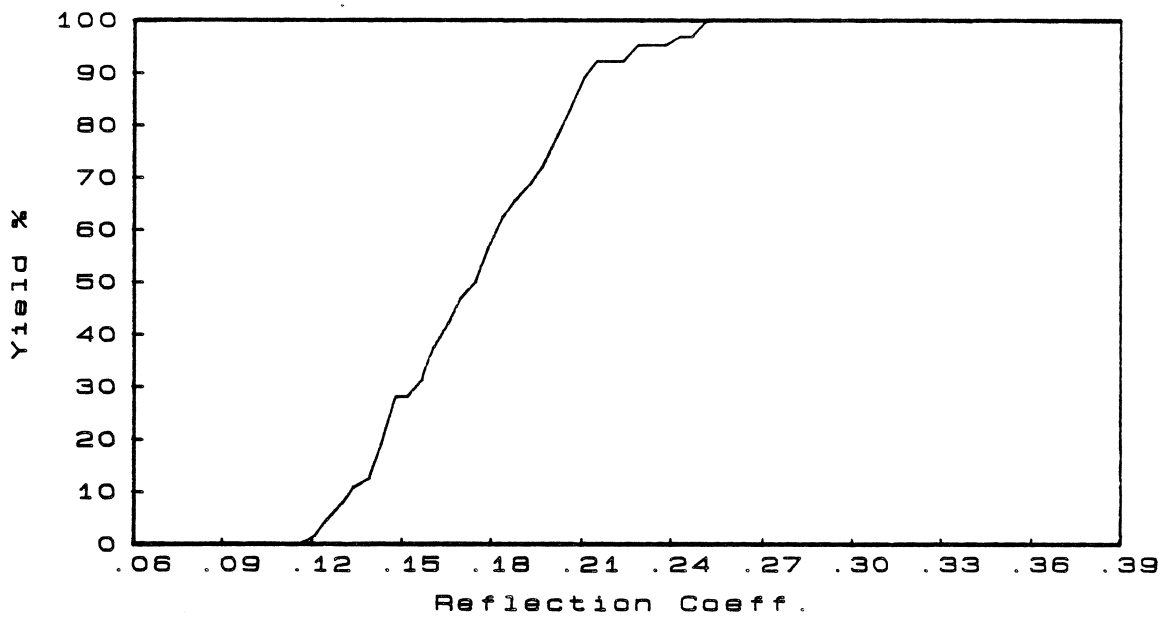
FILE: VA1N64
RANGES OF REFLECTION COEFFICIENTS : .1486 .2677

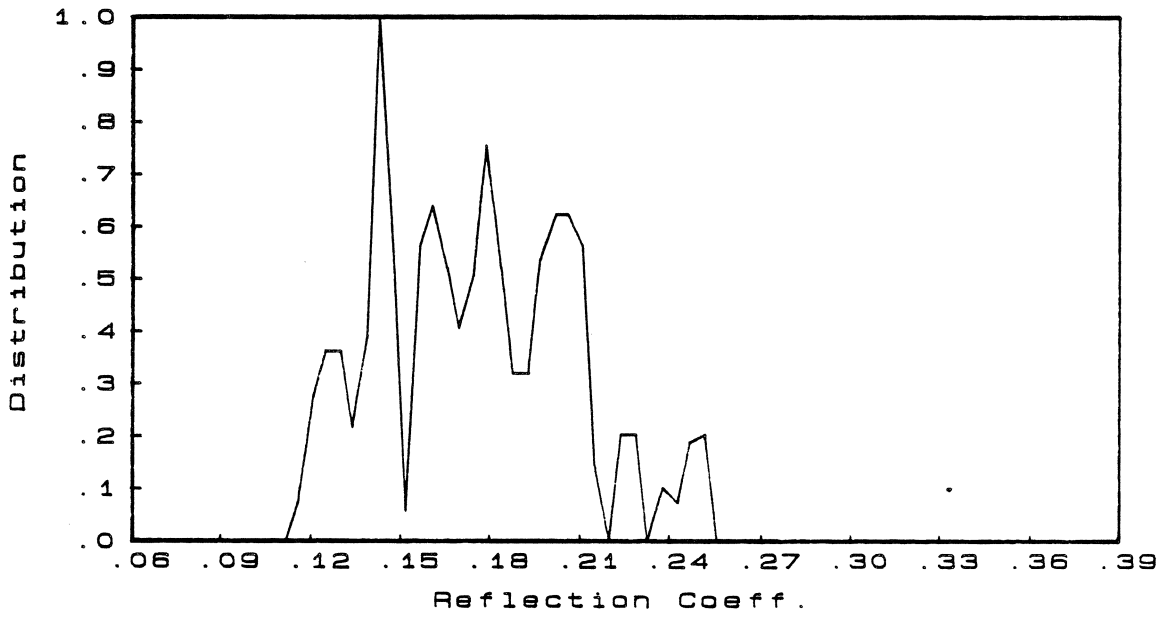
Fig. 8 Vertex analysis of the solution to the FTP obtained using the threshold value of 0.015 for the least squares function characterizing a vertex.



FILE: WKWTMC
RANGES OF REFLECTION COEFFICIENTS : .1107 .1644

Fig. 9 Monte Carlo analysis of the solution to the FTP obtained using weighted vertices plus the nominal point weighted by a factor of 64.





FILE: WKWTVA
RANGES OF REFLECTION COEFFICIENTS : .1205 .2515

Fig. 10 Vertex analysis of the solution to the FTP obtained using weighted vertices plus the nominal point weighted by a factor of 64.

RESULTS FROM THE SPECIALIZED TEST PROGRAM

Introduction

As mentioned in the introduction to this report, constraints imposed by the TOLCAD software, as well as its excessive consumption of CPU time, dictated the need to create a testing program independent of TOLCAD. The specialized short program developed is hereafter referred to as FTPTST. It basically simulates TOLCAD for specific problems and can be used to verify all results obtained using TOLCAD in the fixed tolerance problem mode of operation. Furthermore, FTPTST is readily modified to implement new formulations of the objective function used in the fixed tolerance problem and could thereby be used to explore various design strategies.

This section presents results obtained for various combinations of cases and formulations. The formulations indicate which fixed tolerance design was performed.

Formulations

For completeness, all objective functions and formulations used are explicitly defined. Consistency of notation is maintained within this section.

Zero Tolerance Nominal Design

The following objective function is defined:

$$U = \frac{1}{NF} \sum_{f=FI}^{FF} (RHO^{f}) **2, \qquad (1)$$

where

U is the scalar objective function

NF is the number of sample frequencies

FI is the first sample frequency

FF is the last sample frequency

RHO^f is the reflection coefficient at frequency f

Given starting point variable values of all 1.0 the conventional nominal design optimization without tolerances yields the solution

$$\mathbf{x} = [0.32285 \quad 2.8964 \quad 0.76090 \quad 2.2812 \quad 0.96591 \quad 0.96759]^T$$

with corresponding maximum reflection coefficient over the 21 sample frequencies being 0.119.

Formulation 1

Specification on the Sum of Squares Term

The following objective functions were defined.

1. N worst vertices:

$$U = \sum_{p=1}^{N} ERR_p \tag{2}$$

for p going from the 1st to the Nth worst vertex.

2. N worst vertices + weighted nominal:

$$U = \sum_{p=1}^{N} ERR_p + N * ERR_{po} \tag{3}$$

for p going from the 1st to the Nth worst vertex, po representing the nominal point.

3. R random outcomes:

$$U = \sum_{p=1}^{R} ERR_p \tag{4}$$

for p going from the 1st to the Rth random outcome.

For all cases:

$$ERR_p = \begin{cases} 0.0 & \text{if } SUMSQ_p < SPEC \\ (SUMSQ_p - SPEC) & \text{otherwise} \end{cases} \quad (5)$$

$$SUMSQ_p = 1/NF * \sum_{f=FI}^{FF} (RHO_p{}^f) **2 \quad (6)$$

where NF, FI and FF have been defined already and $RHO_p{}^f$ is the reflection coefficient at point p and frequency f.

Formulation 2

Specification on the Reflection Coefficient

The following objective functions were defined.

1. N worst vertices:

$$U = \sum_{p=1}^{N} (1/NUM * \sum_{f=FI}^{FF} (ERR_p{}^f) **2) \quad (6)$$

for p going from the 1st to the Nth worst vertex.

2. N worst vertices + weighted nominal:

$$U = \sum_{p=1}^{N} (1/NUM * \sum_{f=FI}^{FF} (ERR_p{}^f) **2)$$

$$+ \quad 1/NUM * \sum_{f=FI}^{FF} (ERR_{po}{}^f) **2 \quad (7)$$

for p going from the 1st to the Nth worst vertex, po representing the nominal point.

3. R random outcomes:

$$U = \sum_{p=1}^{R} (1/NUM * \sum_{f=FI}^{FF} (ERR_p{}^f) **2) \quad (8)$$

for p going from the 1st to the Rth random outcome.

For all cases

$$ERR_p^f = \begin{cases} 0.0 & \text{if } RHO_p^f < SPEC \\ (RHO_p^f - SPEC) & \text{otherwise} \end{cases}$$

where

RHO_p^f is the reflection coefficient at point p and frequency f

NUM is the number of nonzero ERR_p^f terms at point p or po, as the case may be.

Results from the Specialized Test Program

Results from the specialized test program FTPTST follow. The RESULTS table summarizes the results of Monte Carlo and vertex analyses performed on the design solutions and the FILENAMES table gives a cross reference to the plots of the relevant distributions. The OPTIMIZATION ITERATION COUNTS table summarizes the computational effort required by the various designs. This effort is measured in terms of number of calls to a subroutine named ANALYZ rather than in seconds since the circuit simulation routine in FTPTST is very fast, whereas the equivalent routine ANALYZ of TOLCAD is the major bottleneck of any design attempt. It is of course possible to correlate design time required by TOLCAD to the number of calls to ANALYZ by using a FTPTST run which duplicates a TOLCAD run. For example, the run which specified the FTPAQ option in TOLCAD was reproduced by FTPTST for the case which considers N=64 worst vertices and SPEC=0.0.

Figures 1 to 24 show curves similar to those obtained as a post analysis to the TOLCAD results. Appendix B lists the associated data.

RESULTS

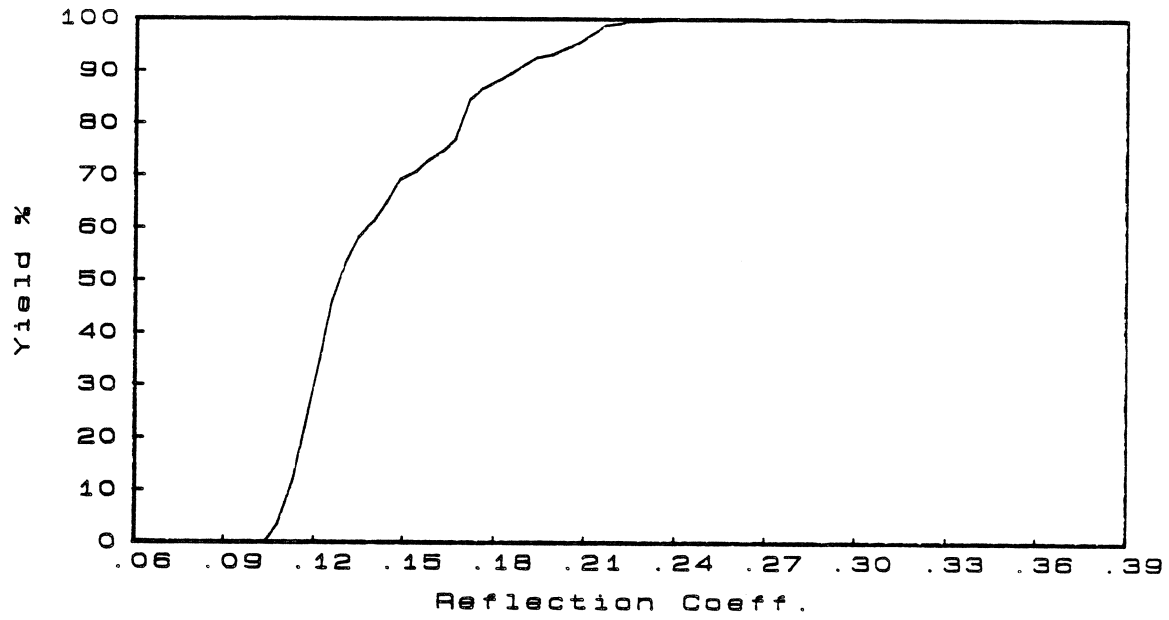
CASE	FORMULATION		MONTE CARLO DISTRIBUTION	VERTEX DISTRIBUTION
Zero Tolerance Nominal Design			.1037 - .2351	.1078 - .3178
N Worst Vertices	SPEC=0.0	N=64	.1295 - .1848	.1192 - .2450
		N=20	.1891 - .2362	.1763 - .2500
	SPEC=0.015 on sum sq. (Form. 1)	N=64	.1626 - .2146	.1487 - .2677
		N=20	.1668 - .2187	.1538 - .2594
	SPEC=0.119 on rho (Form. 2)	N=64	.1011 - .1601	.1009 - .2231
		N=20	.1256 - .1828	.1254 - .2142
N Worst Vertices + Weighted Nominal (wt=N)	SPEC=0.0	N=64	.1107 - .1644	.1205 - .2515
		N=20	.1282 - .1834	.1221 - .2472
R Random Outcomes	SPEC=0.0	R=50	.1082 - .1724	.1167 - .2533
	SPEC=0.002 on sum sq. (Form. 1)	R=50	.1081 - .1724	.1166 - .2532
	SPEC=0.119 on rho (Form. 2)	R=50	.1084 - .1613	.1097 - .2281

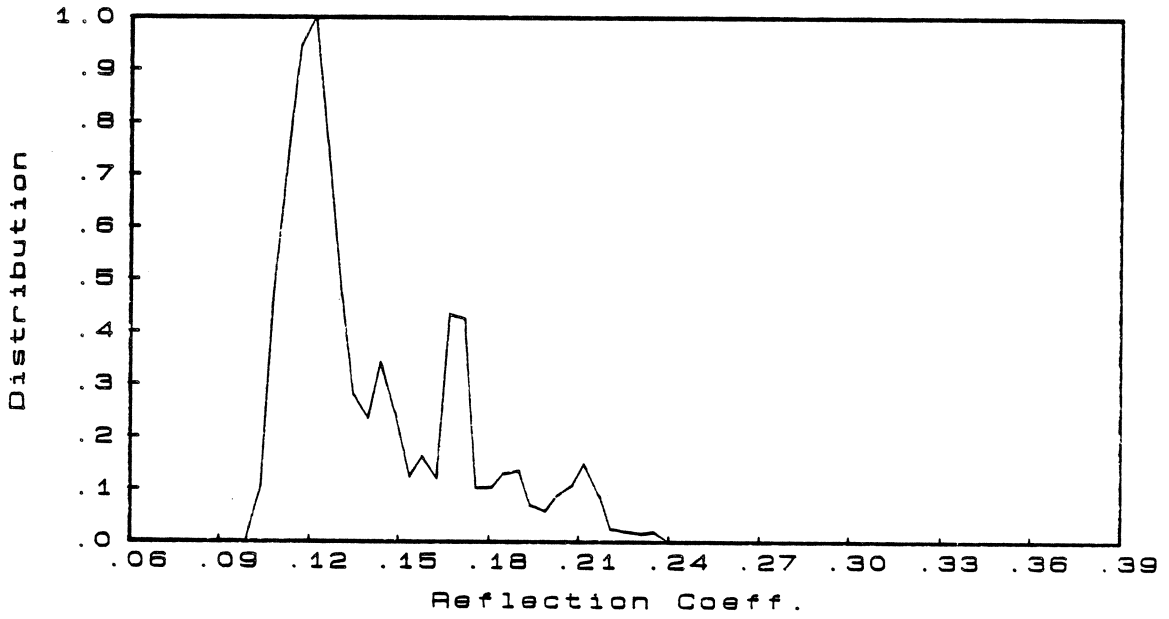
ITERATION AND FUNCTION EVALUATION COUNTS

CASE	FORMULATION		NUMBER OF ITERATIONS	NUMBER OF CALLS TO ANALYZ
Zero Tolerance Nominal Design			25	31
N Worst Vertices	SPEC=0.0	N=64	9	1088
		N=20	17	564
	SPEC=0.015 on sum sq. (Form. 1)	N=64	27	5568
		N=20	26	804
	SPEC=0.119 on rho (Form. 2)	N=64	16	3264
		N=20	10	544
N Worst Vertices + Weighted Nominal (wt=N)	SPEC=0.0	N=64	9	1170
		N=20	10	422
R Random Outcomes	SPEC=0.0	R=50	8	750
	SPEC=0.002 on sum sq. (Form. 1)	R=50	9	800
	SPEC=0.119 on rho (Form. 2)	R=50	15	1000

FILENAMES

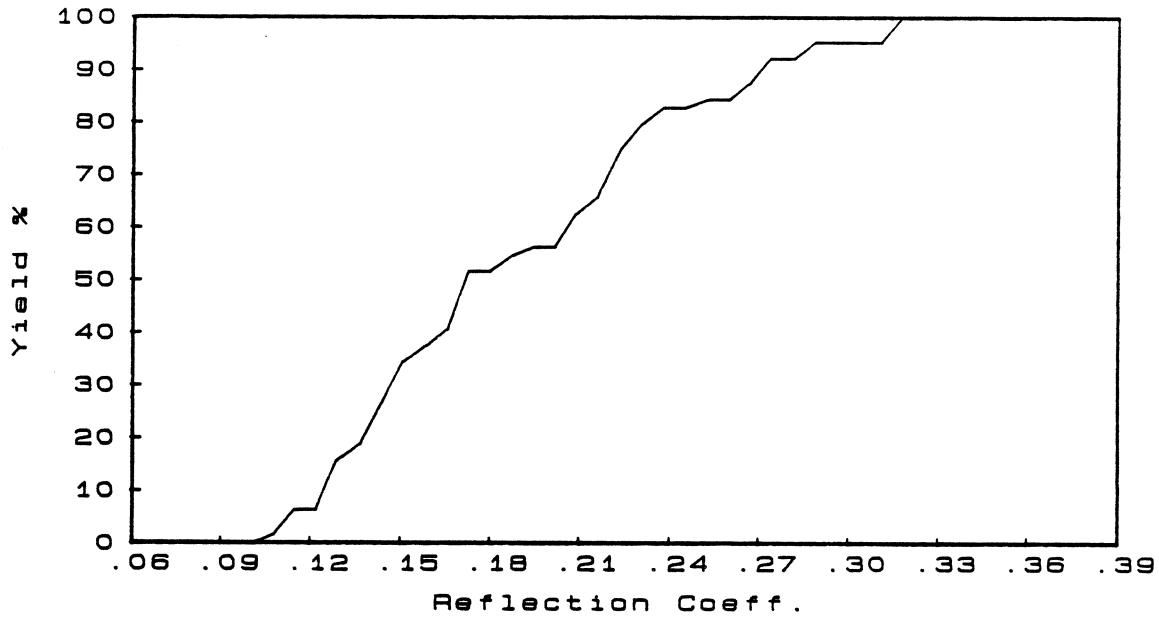
CASE	FORMULATION		MONTE CARLO DISTRIBUTION	VERTEX DISTRIBUTION
Zero Tolerance Nominal Design			MCNOM	VANOM
N Worst Vertices	SPEC=0.0	N=64	MCON64	VAON64
		N=20	MCON20	VAON20
	SPEC=0.015 on sum sq. (Form. 1)	N=64	MC1N64	VA1N64
		N=20	MC1N20	VA1N20
	SPEC=0.119 on rho (Form. 2)	N=64	MC2N64	VA2N64
		N=20	MC2N20	VA2N20
N Worst Vertices + Weighted Nominal (wt=N)	SPEC=0.0	N=64	WMCON64	WVAON64
		N=20	WMCON20	WVAON20
R Random Outcomes	SPEC=0.0	R=50	MC0R50	VA0R50
	SPEC=0.002 on sum sq. (Form. 1)	R=50	MC1R50	VA1R50
	SPEC=0.119 on rho (Form. 2)	R=50	MC2R50	VA2R50

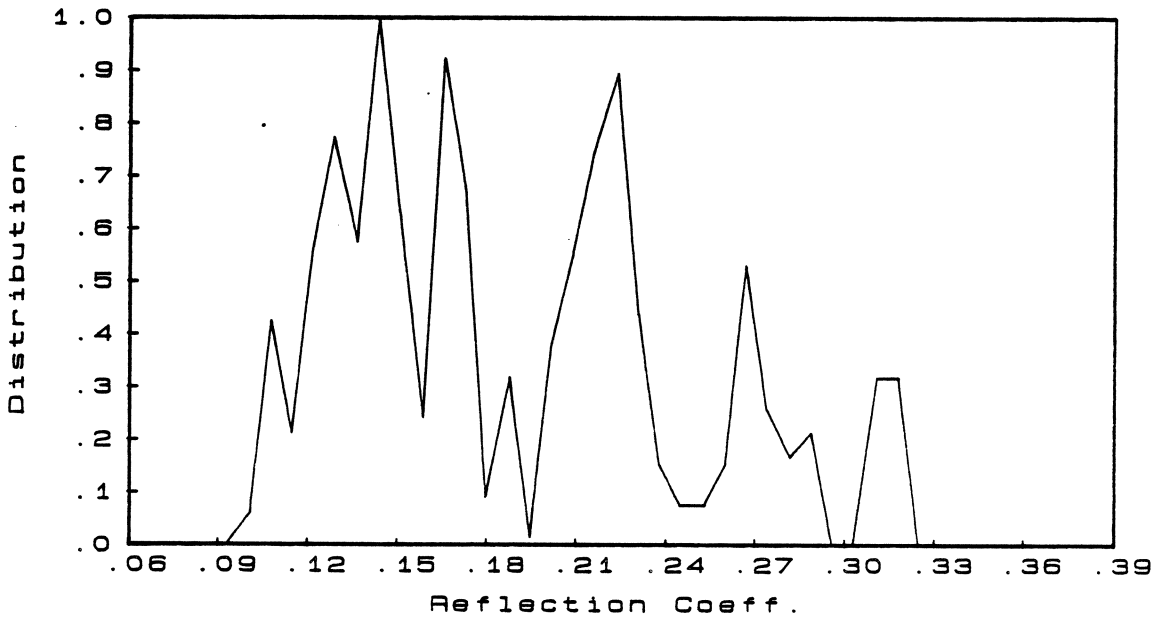




FILE: MCNOM
RANGES OF REFLECTION COEFFICIENTS : .1037 .2351

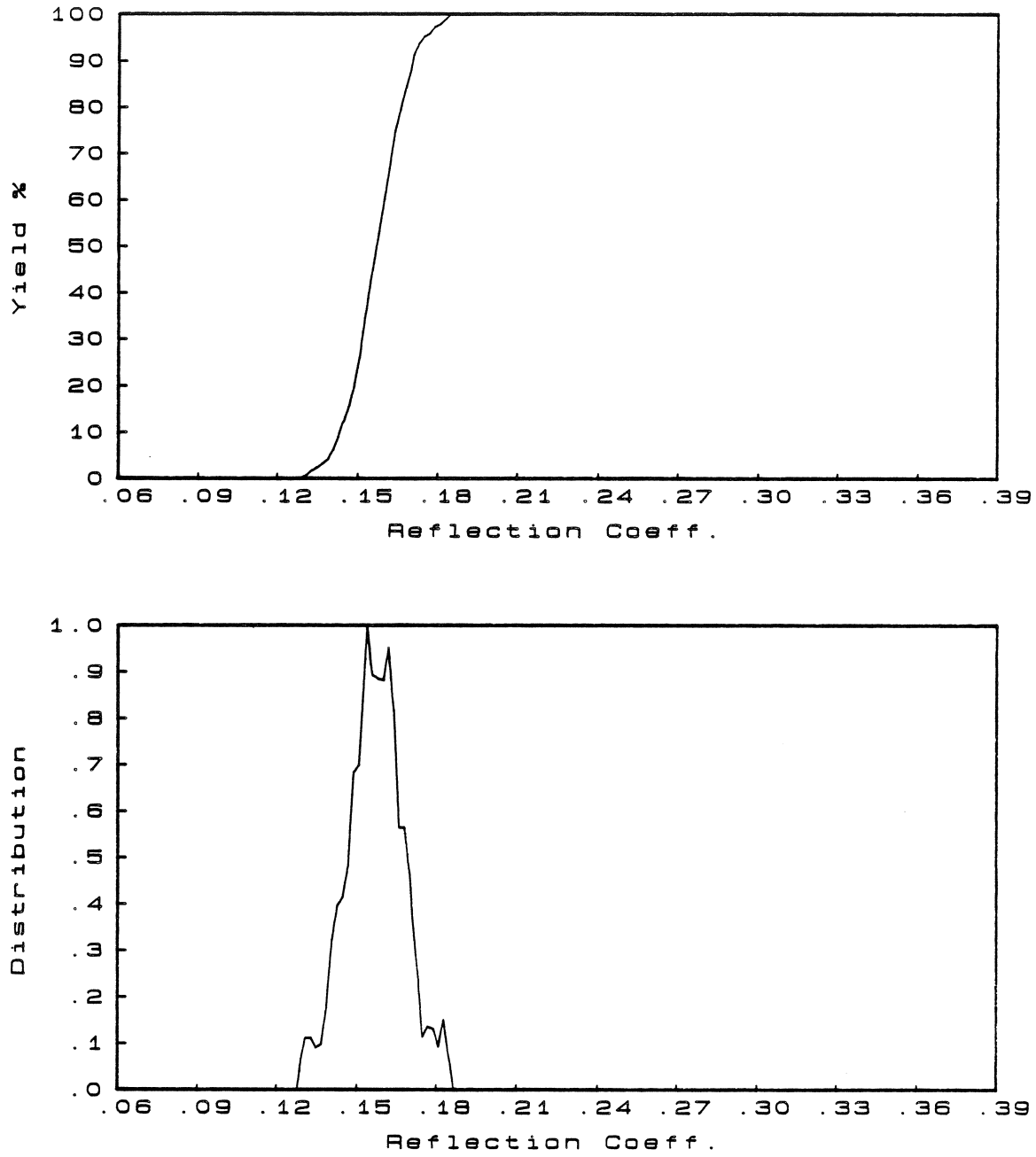
Fig. 1 Monte Carlo analysis of the zero tolerance nominal design with no specification on either the reflection coefficient or the least squares term of the objective function.





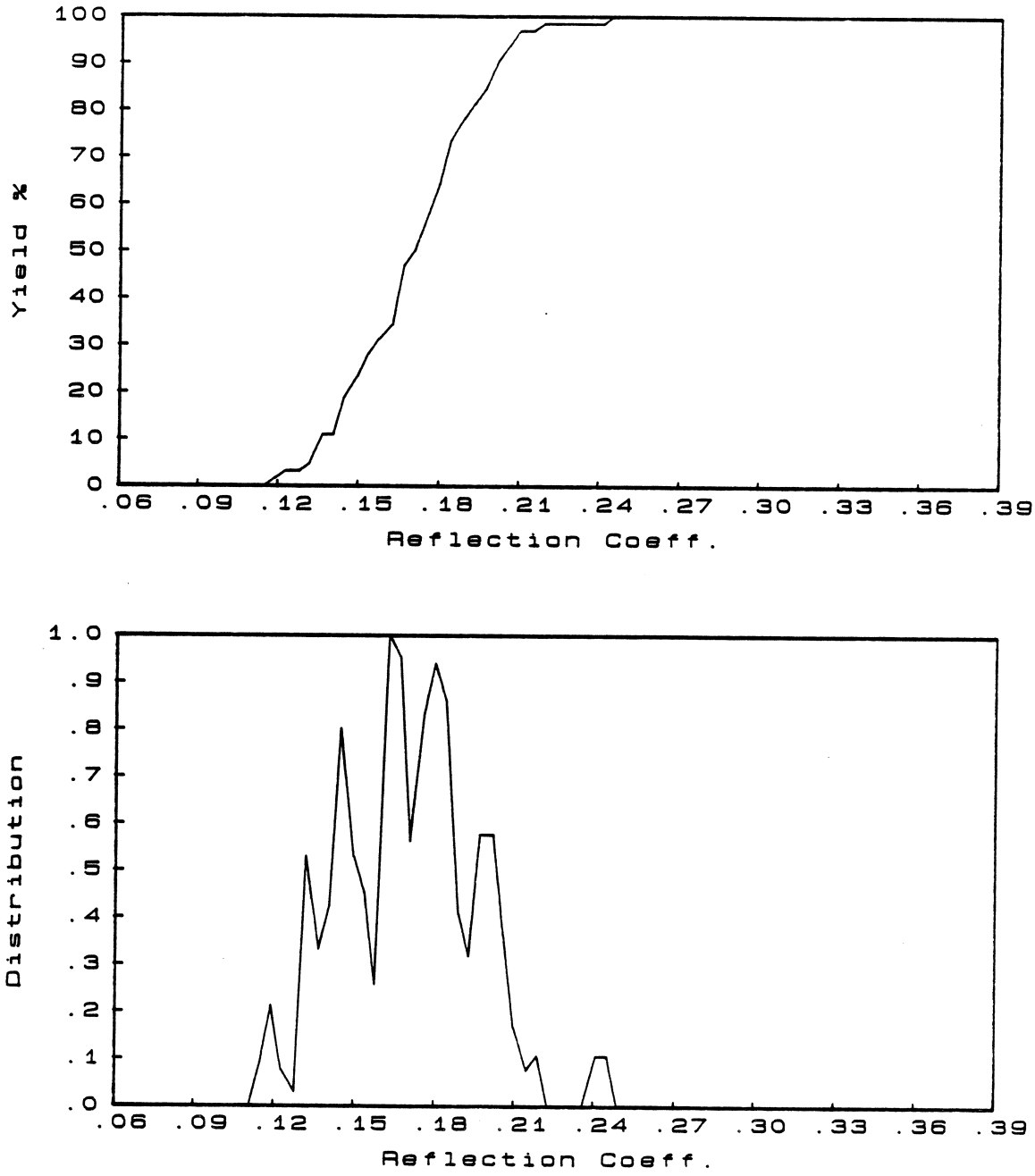
FILE: VANOM
RANGES OF REFLECTION COEFFICIENTS : .1078 .3178

Fig. 2 Vertex analysis of the zero tolerance nominal design with no specification on either the reflection coefficient or the least squares term of the objective function.



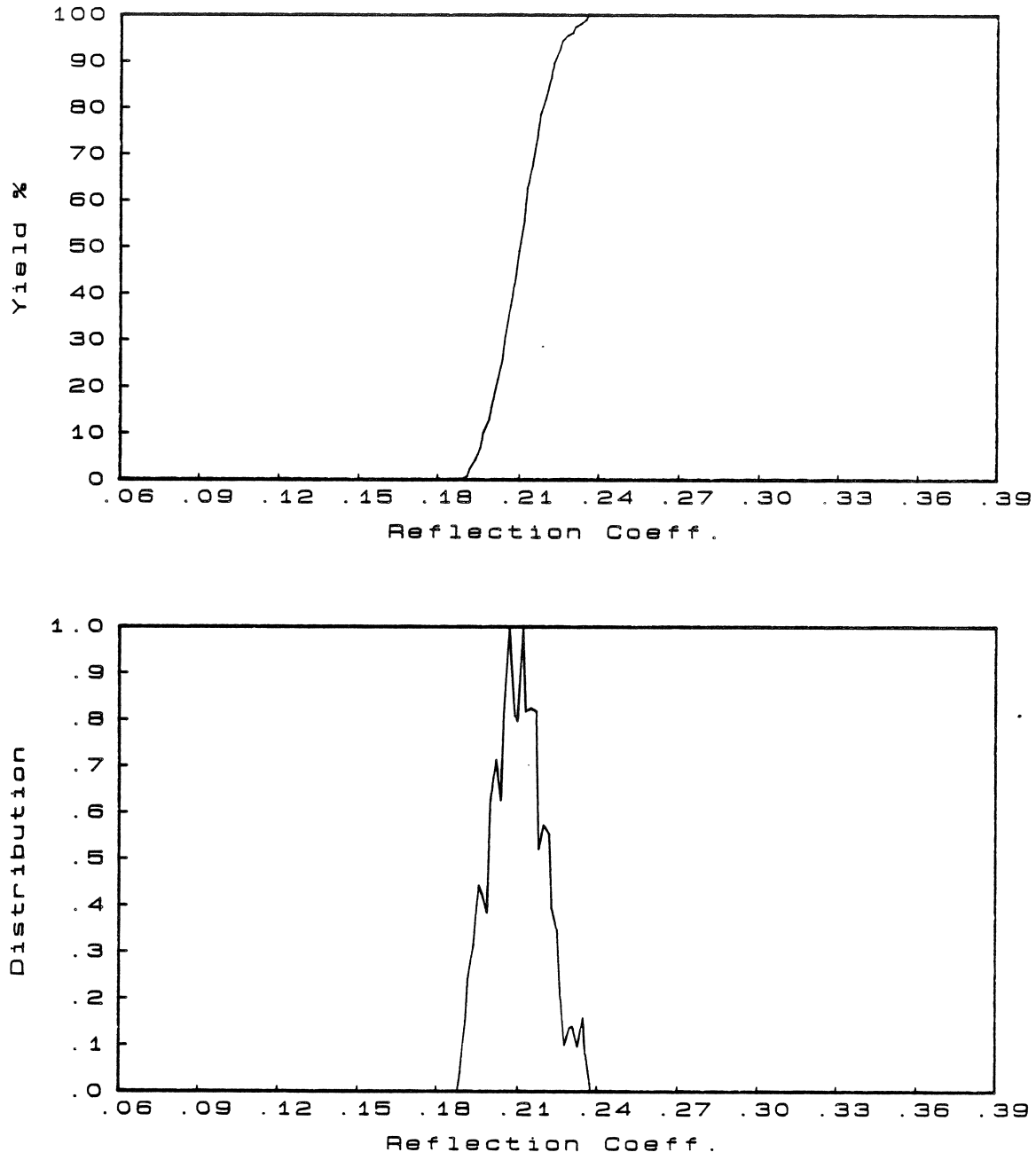
FILE: MCON64
RANGES OF REFLECTION COEFFICIENTS : .1295 .1848

Fig. 3 Monte Carlo analysis of the fixed tolerance design which considered N=64 worst vertices (all) and with no specification on either the reflection coefficient or the least squares term of the objective function.



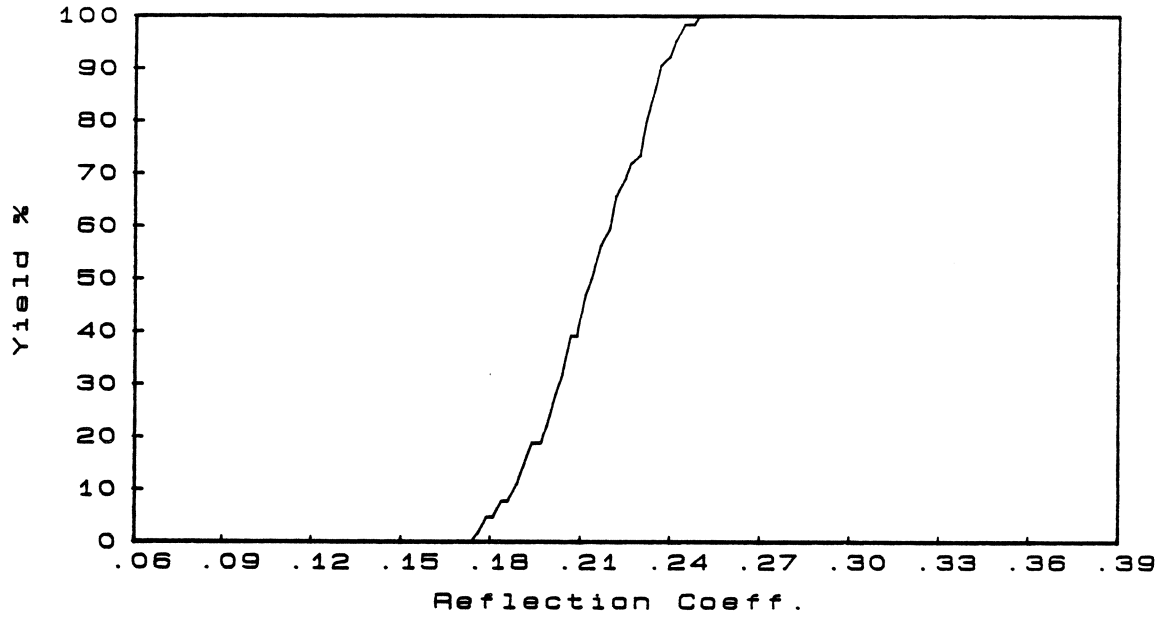
FILE: VAON64
RANGES OF REFLECTION COEFFICIENTS : .1192 .2450

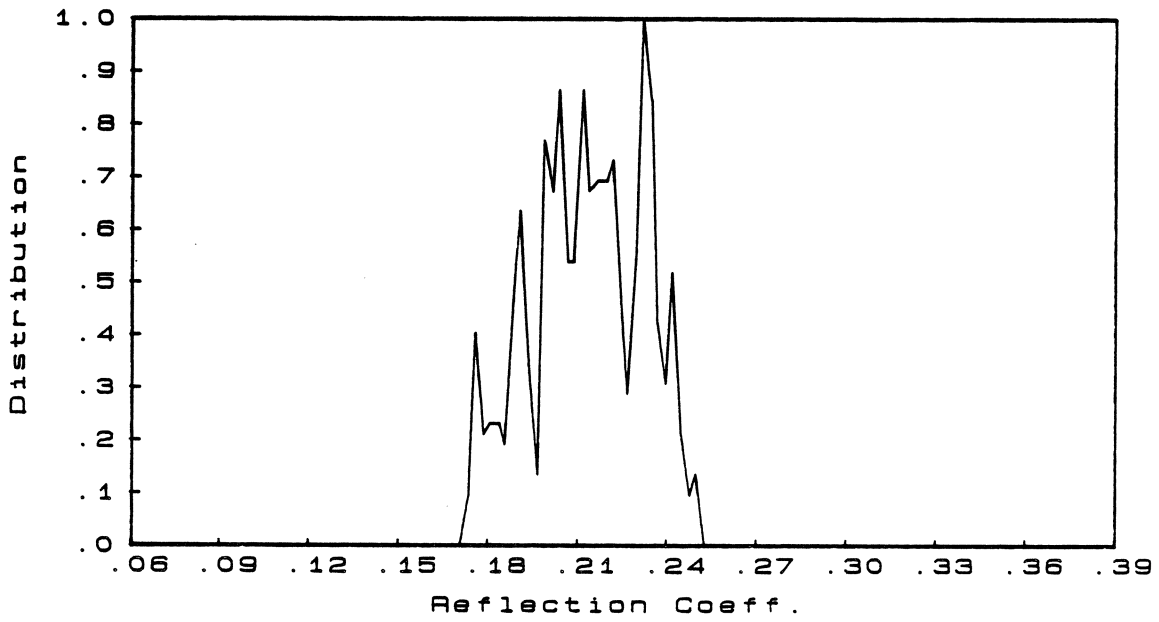
Fig. 4 Vertex analysis of the fixed tolerance design which considered N=64 worst vertices (all) and with no specification on either the reflection coefficient or the least squares term of the objective function.



FILE: MCON20
RANGES OF REFLECTION COEFFICIENTS : .1891 .2362

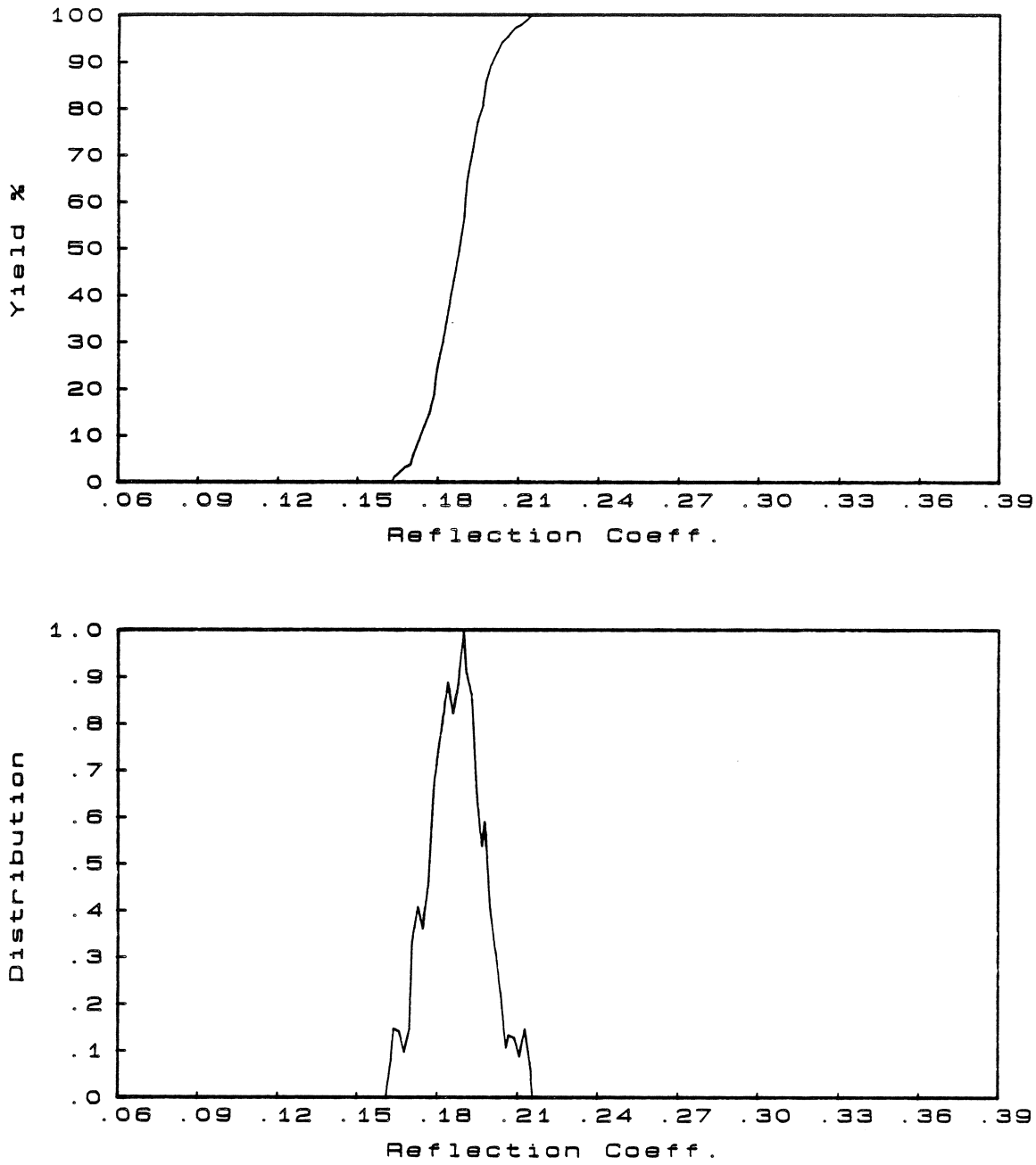
Fig. 5 Monte Carlo analysis of the fixed tolerance design which considered N=20 worst vertices and with no specification on either the reflection coefficient or the least squares term of the objective function.





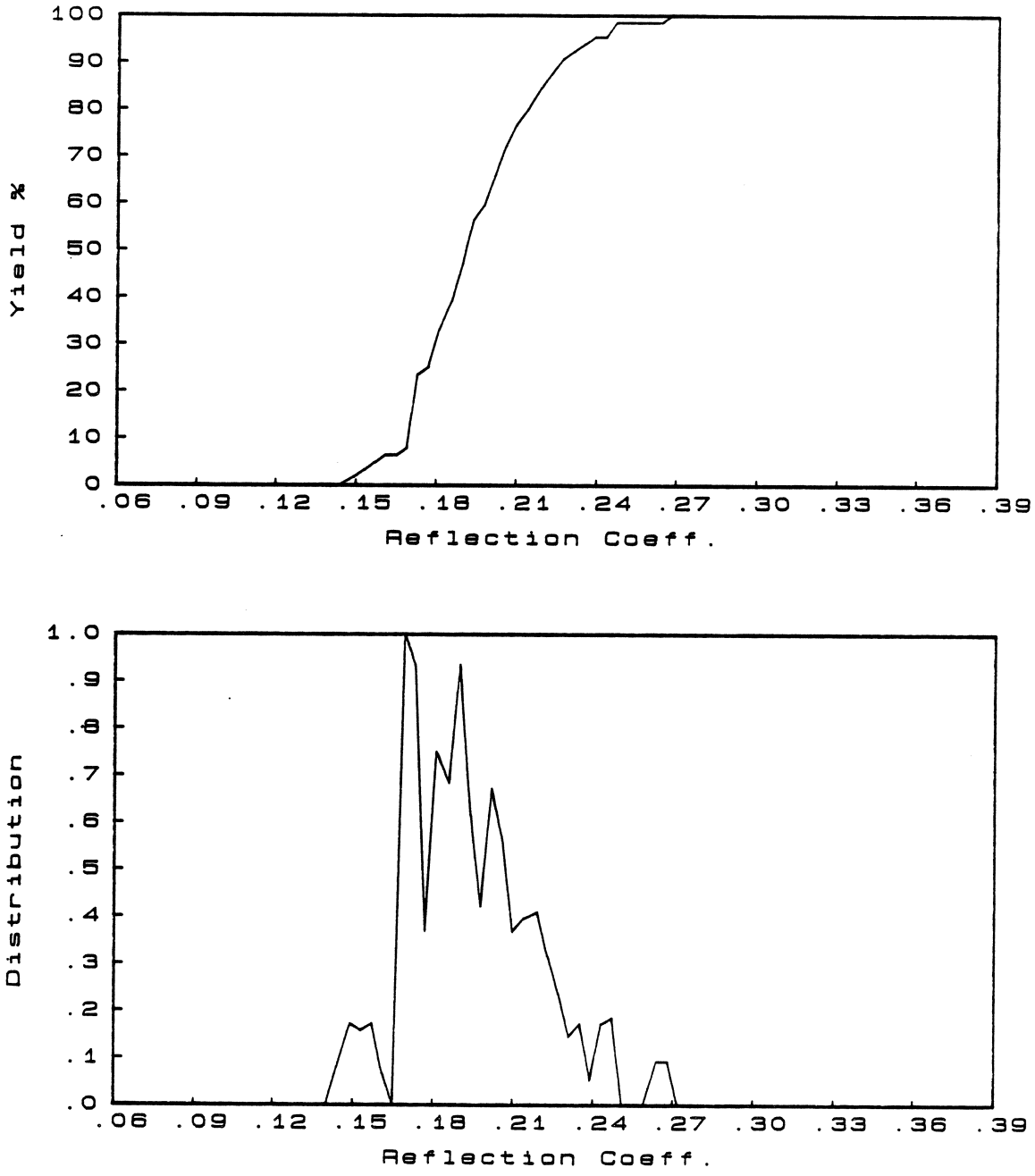
FILE: VAON20
RANGES OF REFLECTION COEFFICIENTS : .1763 .2500

Fig. 6 Vertex analysis of the fixed tolerance design which considered N=20 worst vertices and with no specification on either the reflection coefficient or the least squares term of the objective function.



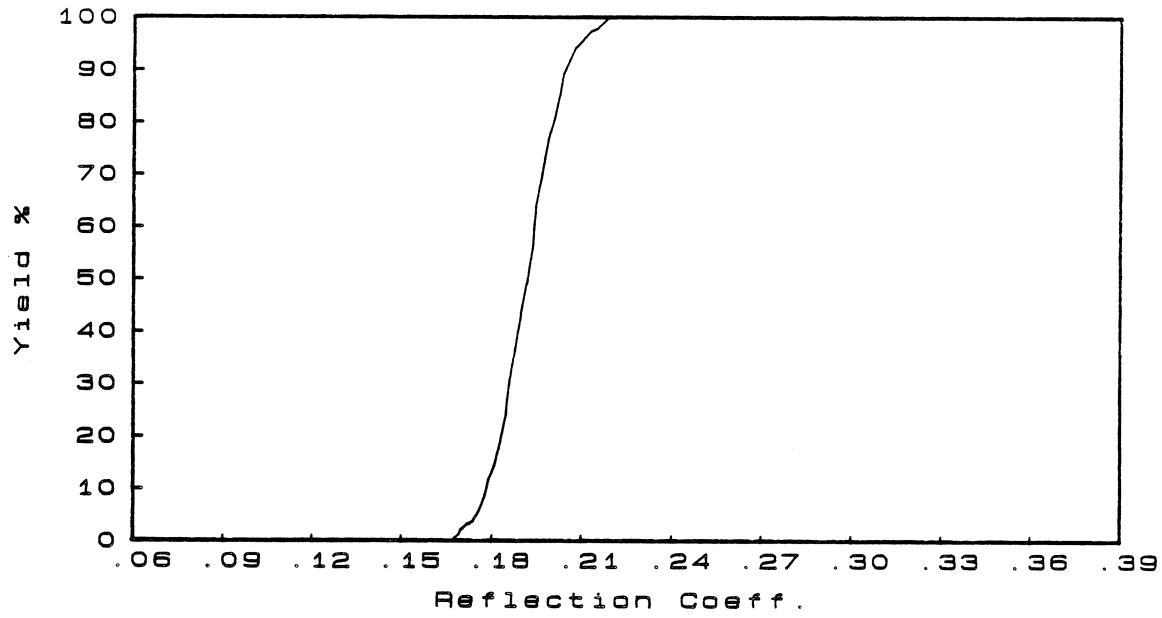
FILE: MC1N64
RANGES OF REFLECTION COEFFICIENTS : .1626 .2146

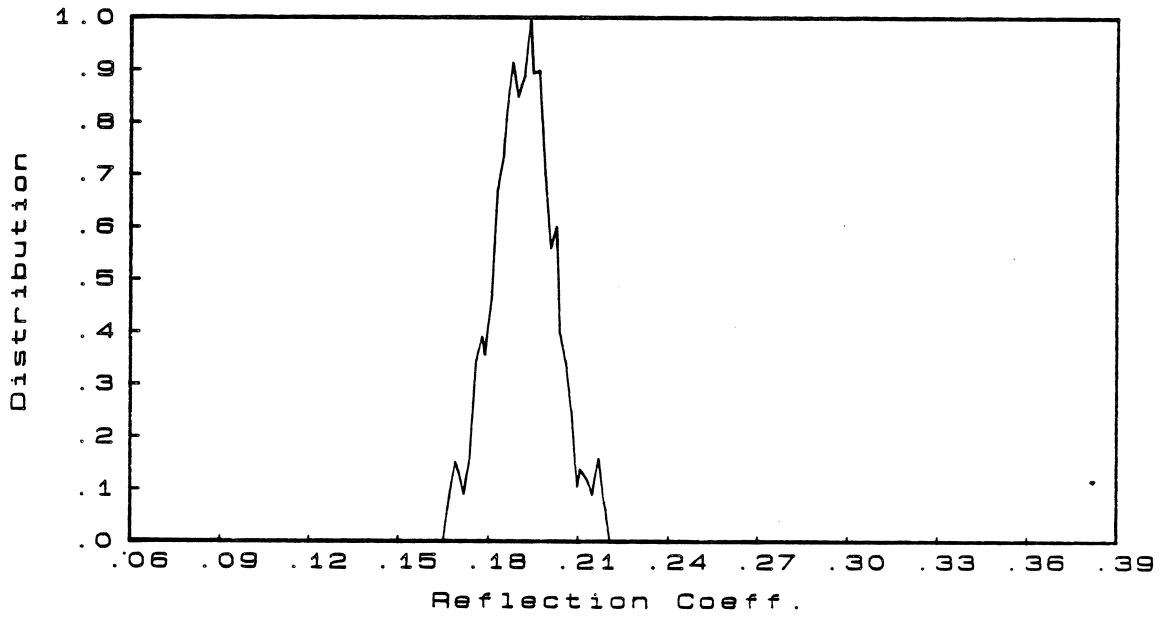
Fig. 7 Monte Carlo analysis of the fixed tolerance design which considered N=64 worst vertices (all) and with a specification value of 0.015 imposed on the least squares term of the objective function.



FILE: VA1N64
RANGES OF REFLECTION COEFFICIENTS : .1487 .2677

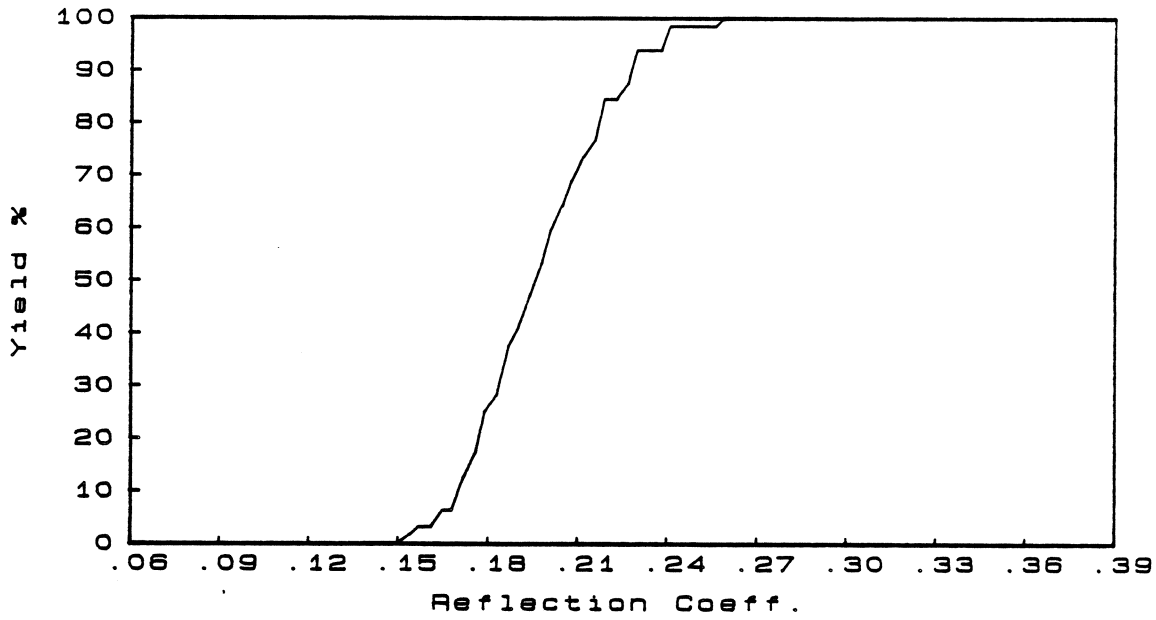
Fig. 8 Vertex analysis of the fixed tolerance design which considered N=64 worst vertices (all) and with a specification value of 0.015 imposed on the least squares term of the objective function.

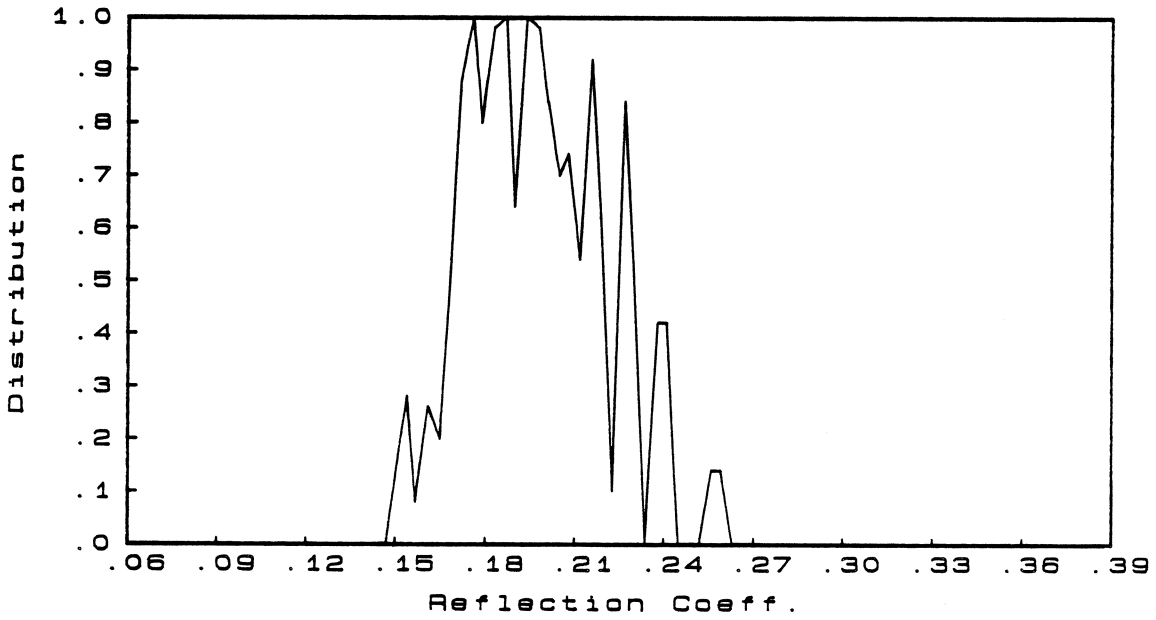




FILE: MC1N20
RANGES OF REFLECTION COEFFICIENTS : .1668 .2187

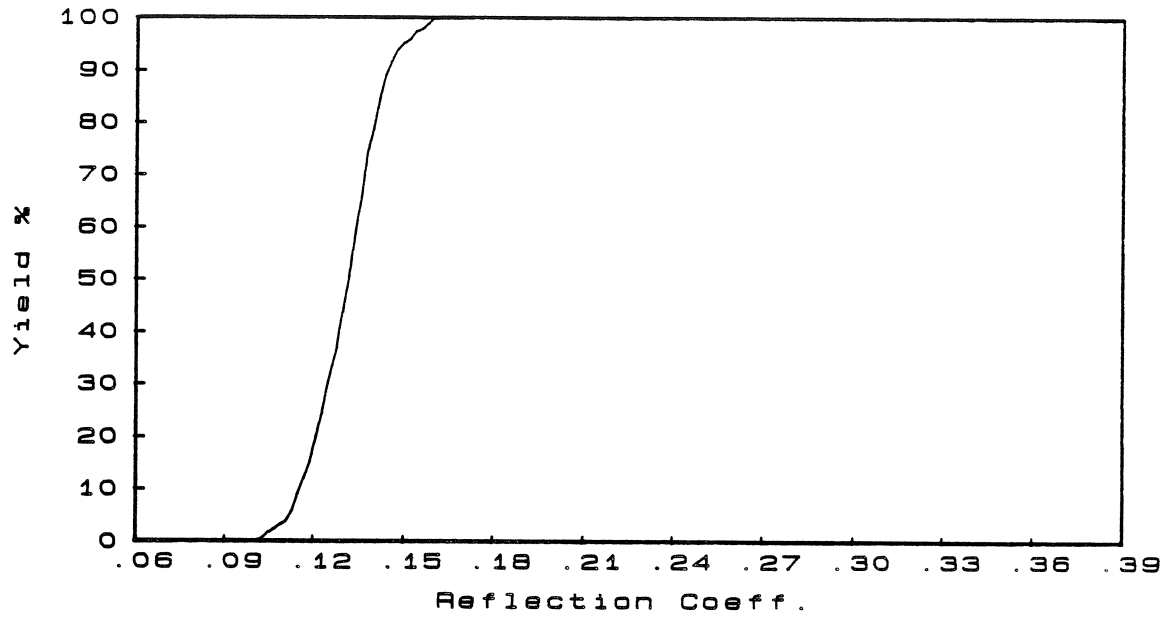
Fig. 9 Monte Carlo analysis of the fixed tolerance design which considered N=20 worst vertices and with a specification value of 0.015 imposed on the least squares term of the objective function.

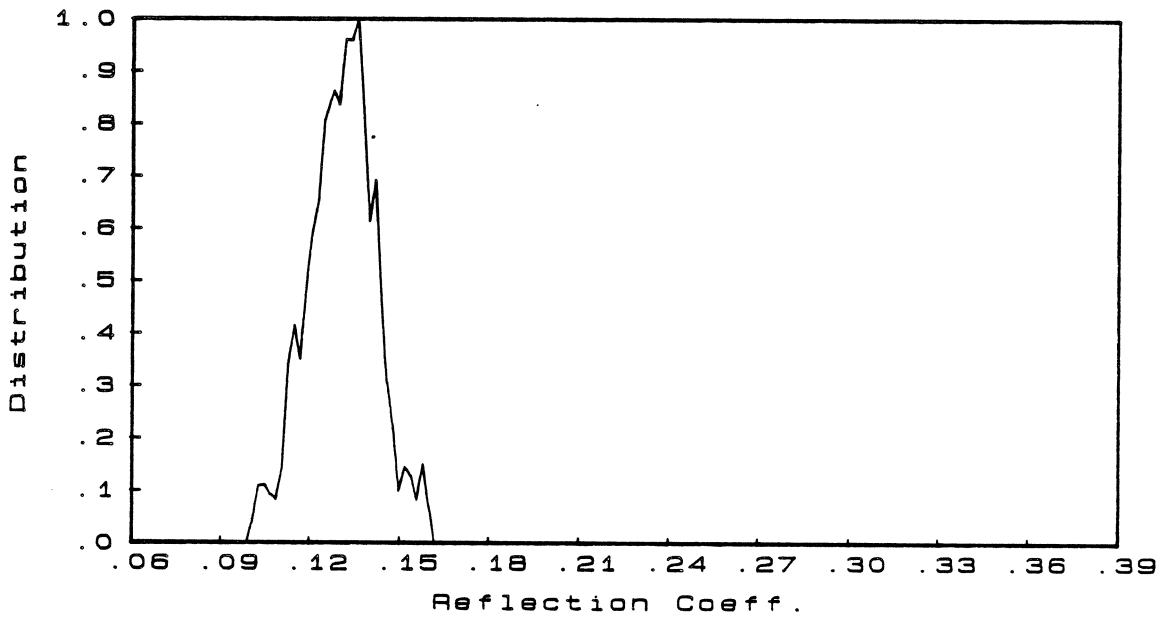




FILE: VA1N20
RANGES OF REFLECTION COEFFICIENTS : .1538 .2594

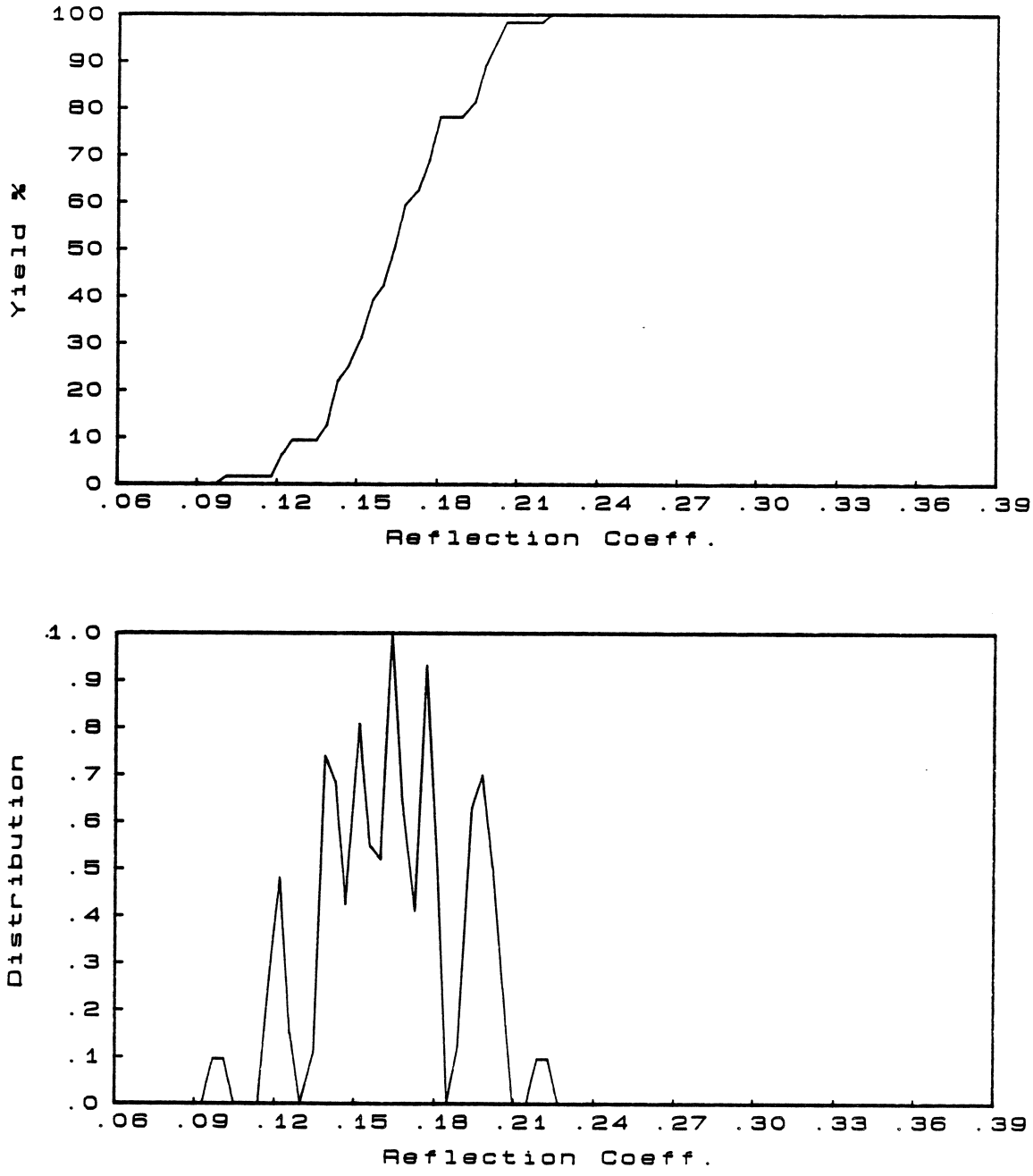
Fig. 10 Vertex analysis of the fixed tolerance design which considered N=20 worst vertices and with a specification value of 0.015 imposed on the least squares term of the objective function.





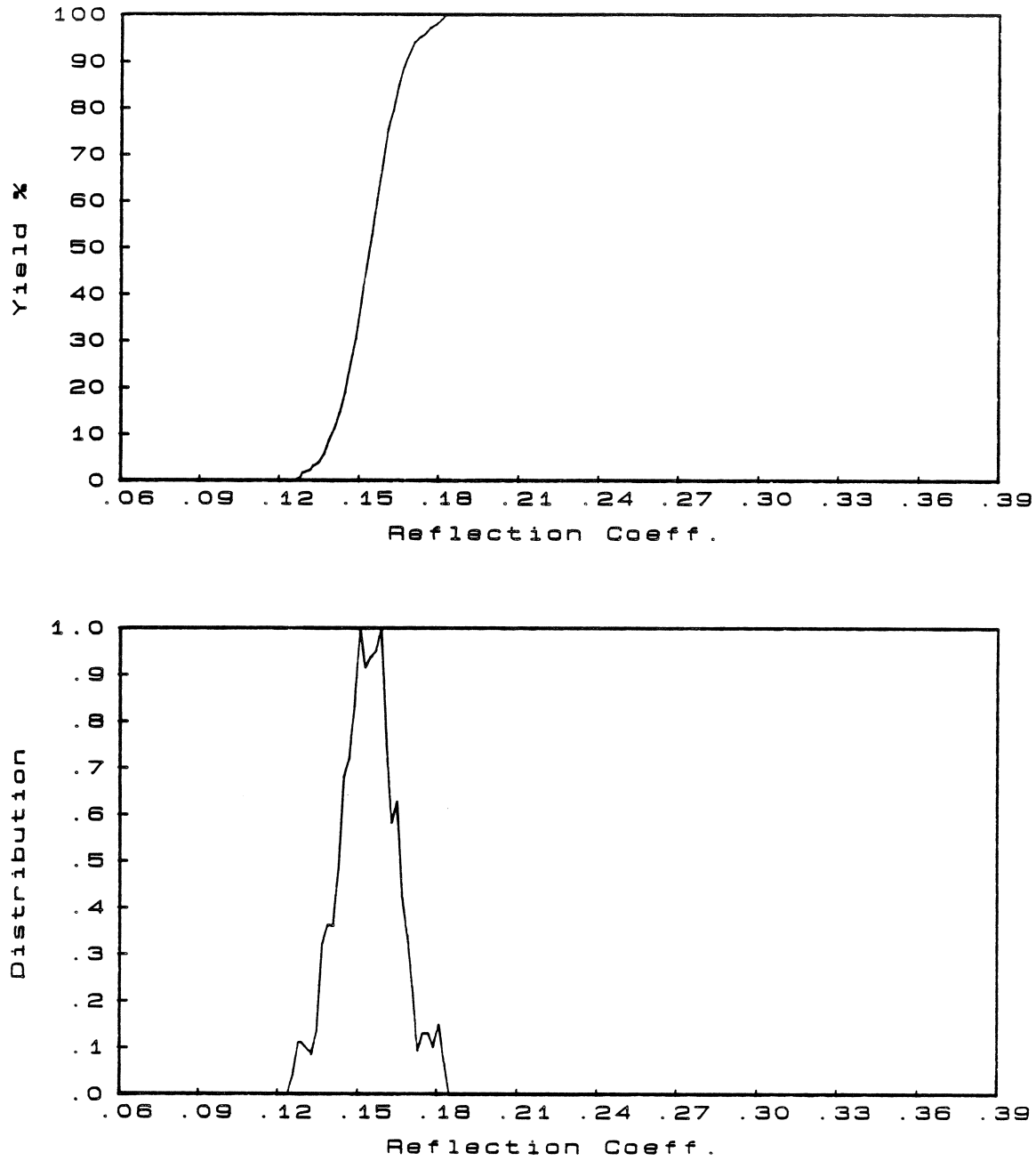
FILE: MC2N64
RANGES OF REFLECTION COEFFICIENTS : .1011 .1601

Fig. 11 Monte Carlo analysis of the fixed tolerance design which considered N=64 worst vertices (all) and with a specification value of 0.119 imposed on each of the reflection coefficients of the objective function.



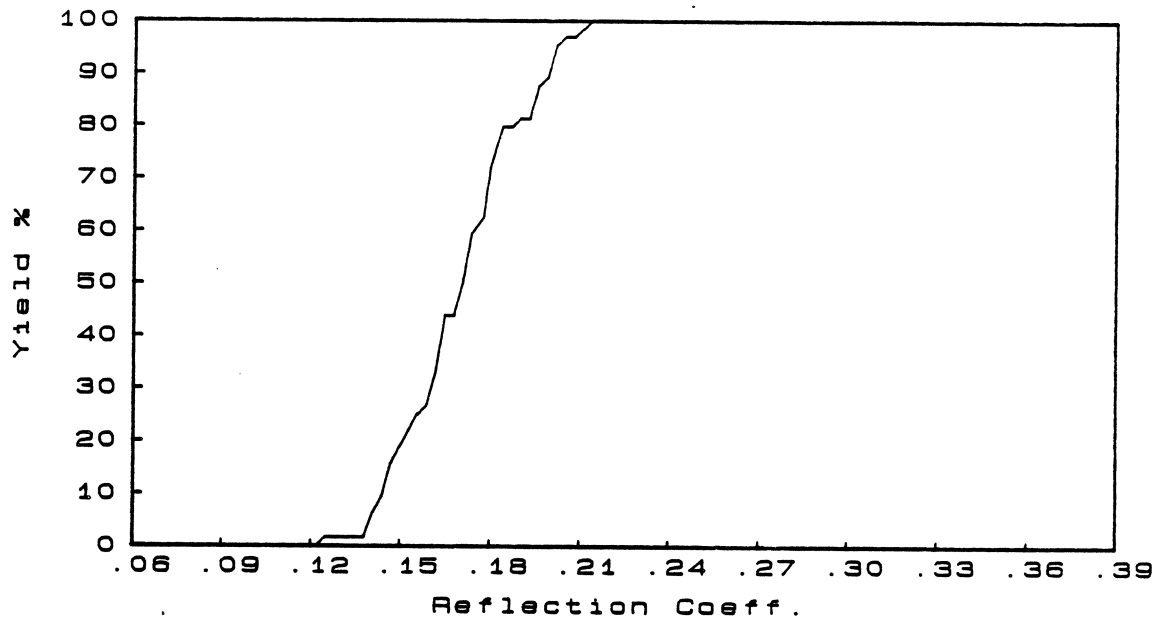
FILE: VA2N64
RANGES OF REFLECTION COEFFICIENTS : .1009 .2231

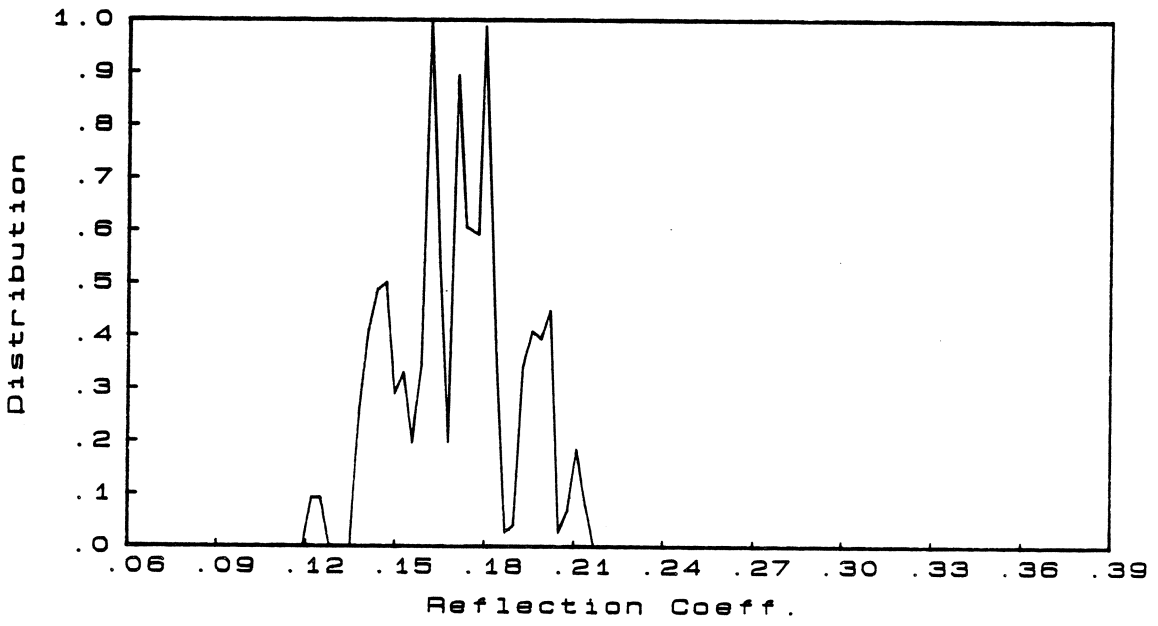
Fig. 12 Vertex analysis of the fixed tolerance design which considered N=64 worst vertices (all) and with a specification value of 0.119 imposed on each of the reflection coefficients of the objective function.



FILE: MC2N20
RANGES OF REFLECTION COEFFICIENTS : .1256 .1828

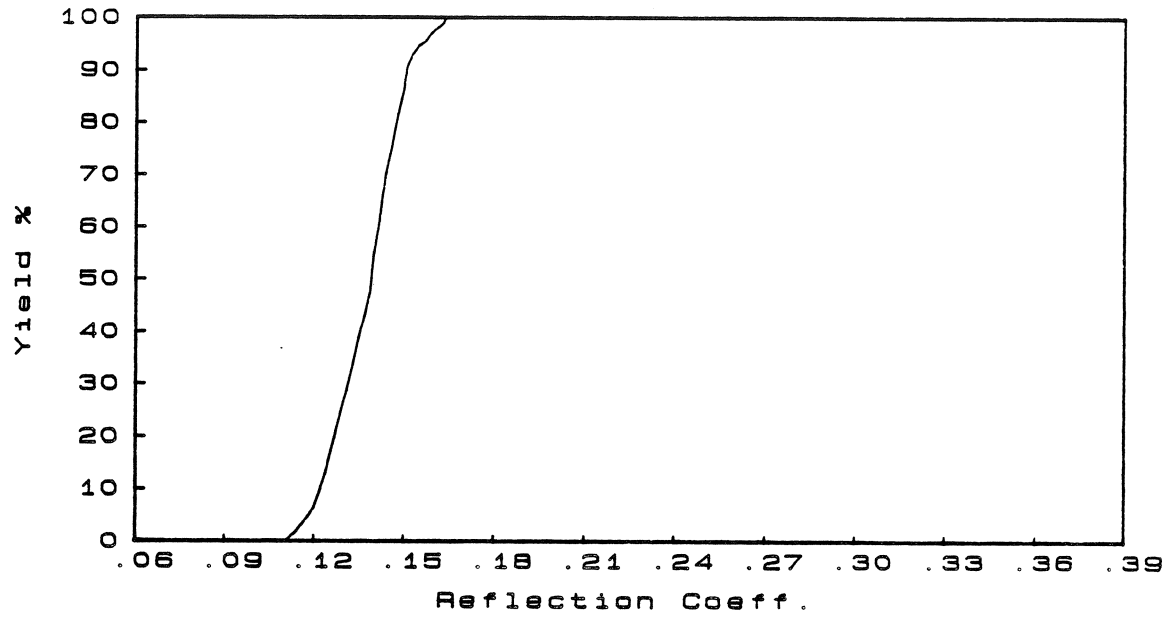
Fig. 13 Monte Carlo analysis of the fixed tolerance design which considered N=20 worst vertices and with a specification value of 0.119 imposed on each of the reflection coefficients of the objective function.

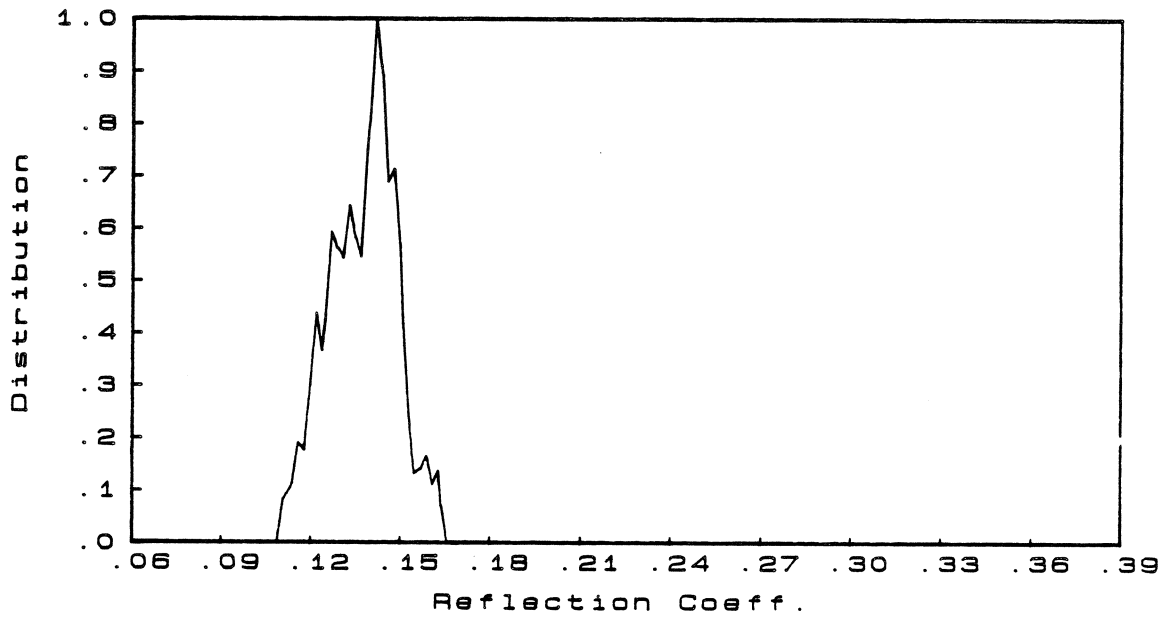




FILE: VA2N20
RANGES OF REFLECTION COEFFICIENTS : .1254 .2142

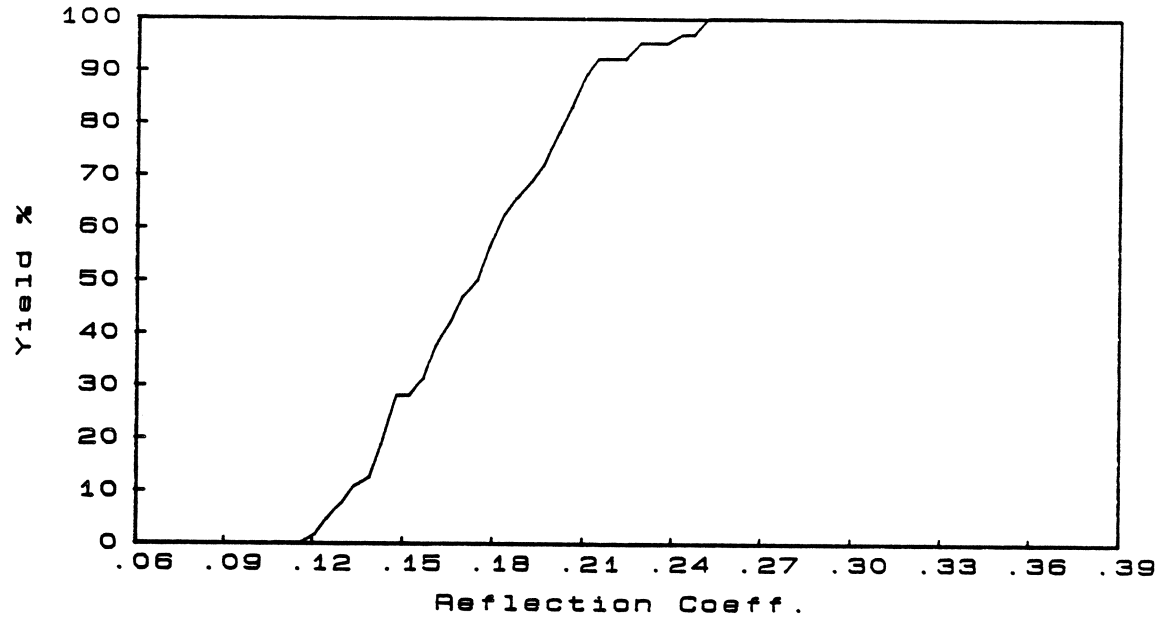
Fig. 14 Vertex analysis of the fixed tolerance design which considered N=20 worst vertices and with a specification value of 0.119 imposed on each of the reflection coefficients of the objective function.

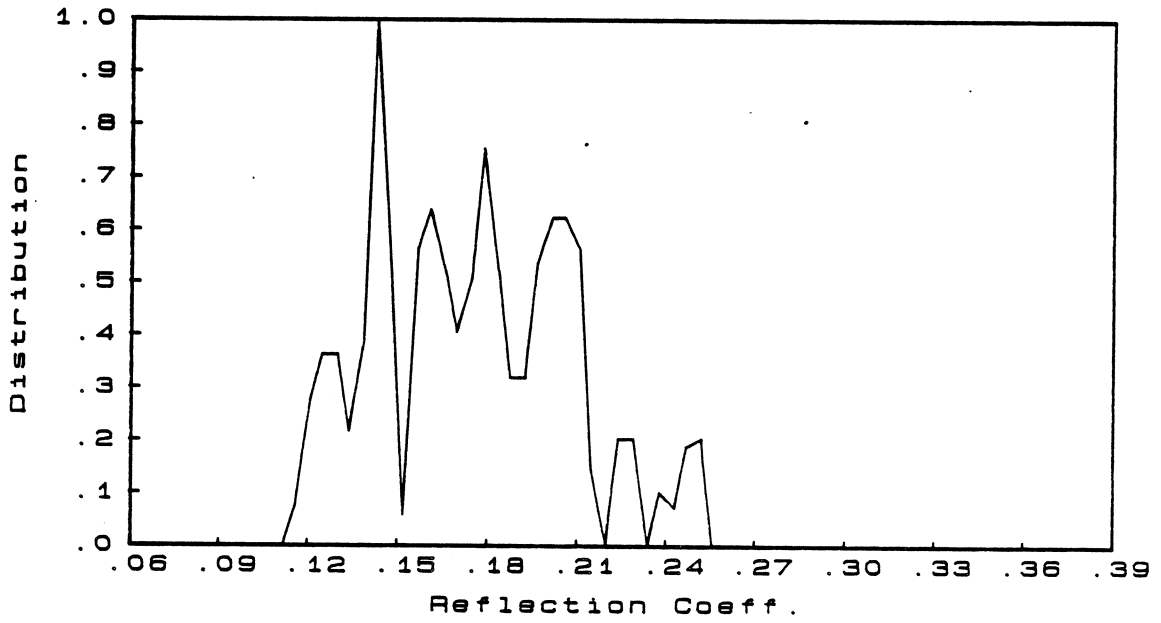




FILE: WMCON64
RANGES OF REFLECTION COEFFICIENTS : .1107 .1644

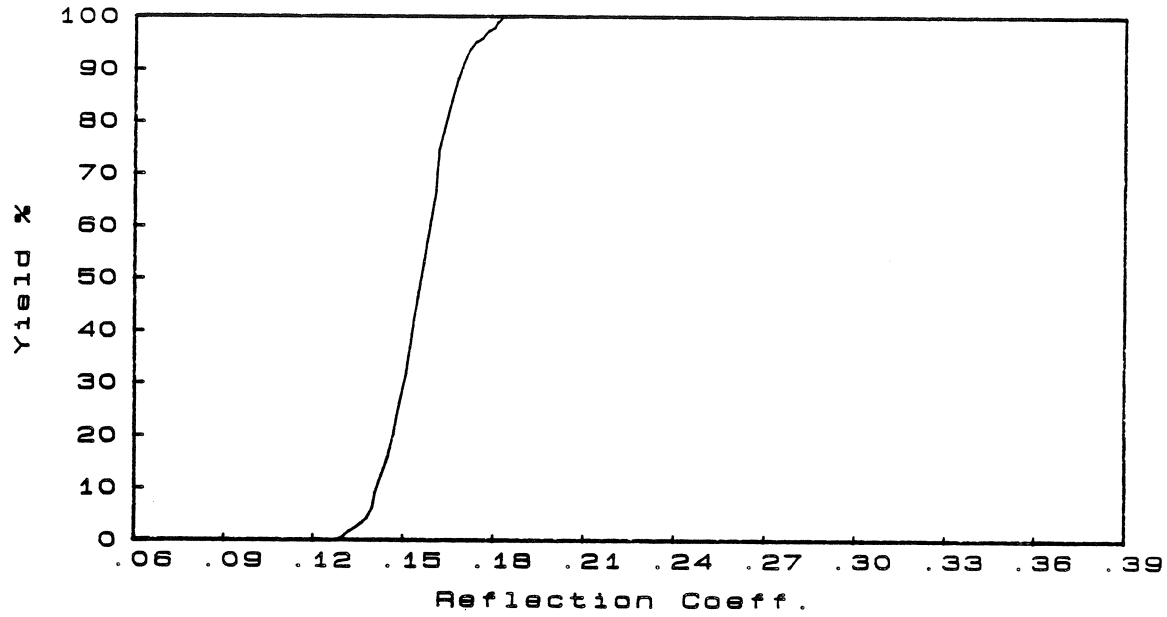
Fig. 15 Monte Carlo analysis of the fixed tolerance design which considered N=64 worst vertices (all) and the weighted nominal point (weight factor=64) and with no specification on either the reflection coefficient or the least squares term of the objective function.

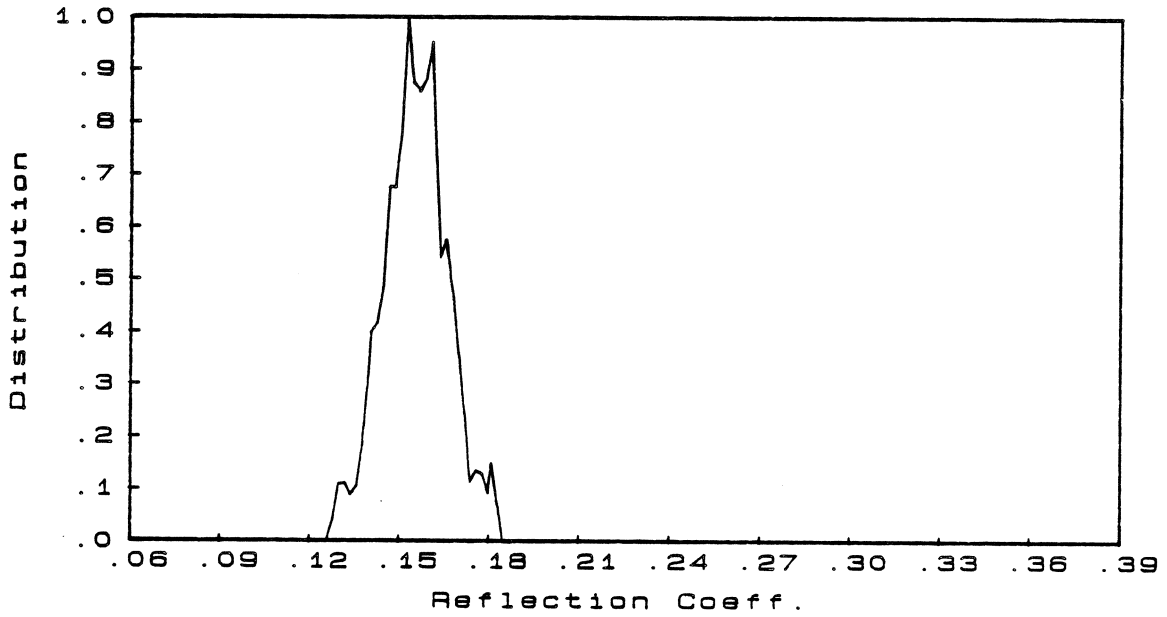




FILE: WVAON64
RANGES OF REFLECTION COEFFICIENTS : .1205 .2515

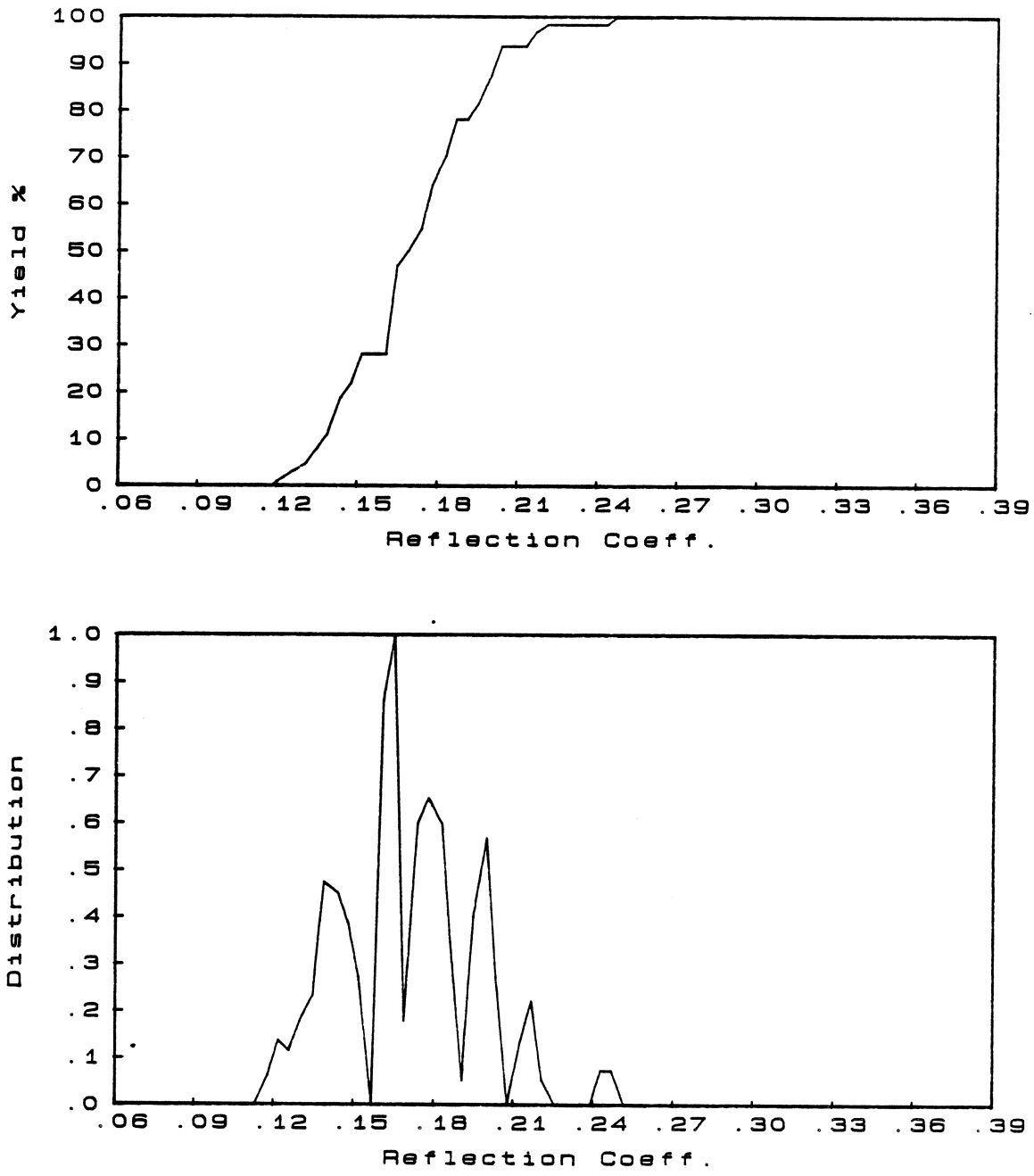
Fig. 16 Vertex analysis of the fixed tolerance design which considered N=64 worst vertices (all) and the weighted nominal point (weight factor=64) and with no specification on either the reflection coefficient or the least squares term of the objective function.





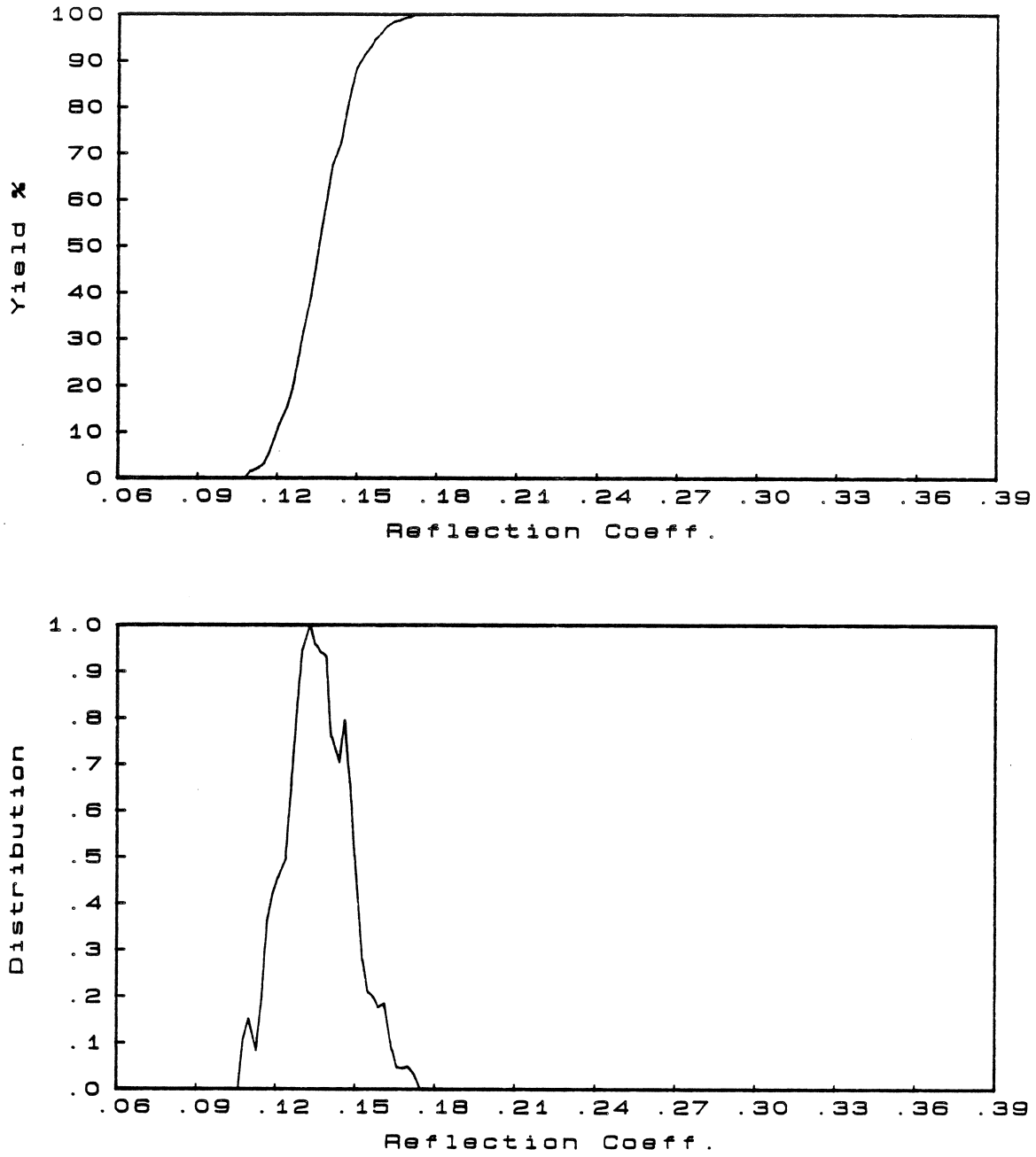
FILE: WMCON20
RANGES OF REFLECTION COEFFICIENTS : .1282 .1834

Fig. 17 Monte Carlo analysis of the fixed tolerance design which considered N=20 worst vertices and the weighted nominal point (weight factor=20) and with no specification on either the reflection coefficient or the least squares term of the objective function.



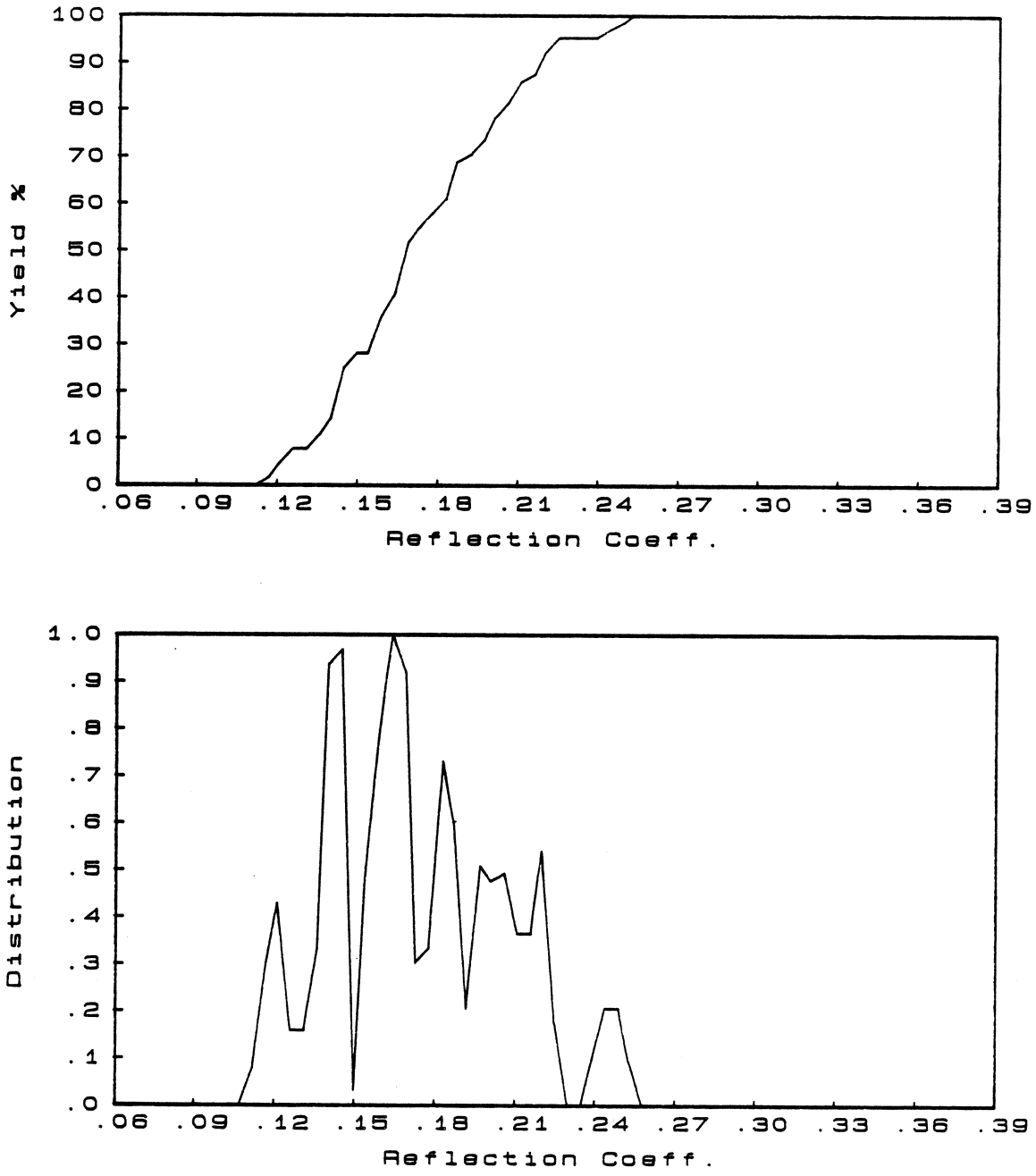
FILE: WVAON20
RANGES OF REFLECTION COEFFICIENTS : .1221 .2472

Fig. 18 Vertex analysis of the fixed tolerance design which considered N=20 worst vertices and the weighted nominal point (weight factor=20) and with no specification on either the reflection coefficient or the least squares term of the objective function.



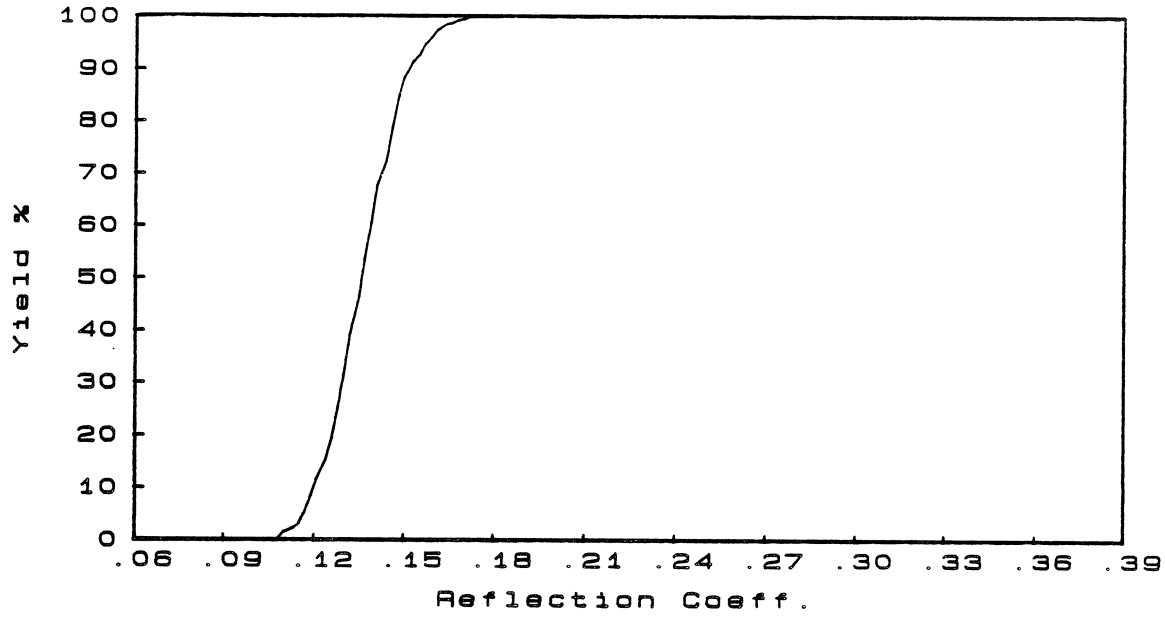
FILE: MCOR50
RANGES OF REFLECTION COEFFICIENTS : .1082 .1724

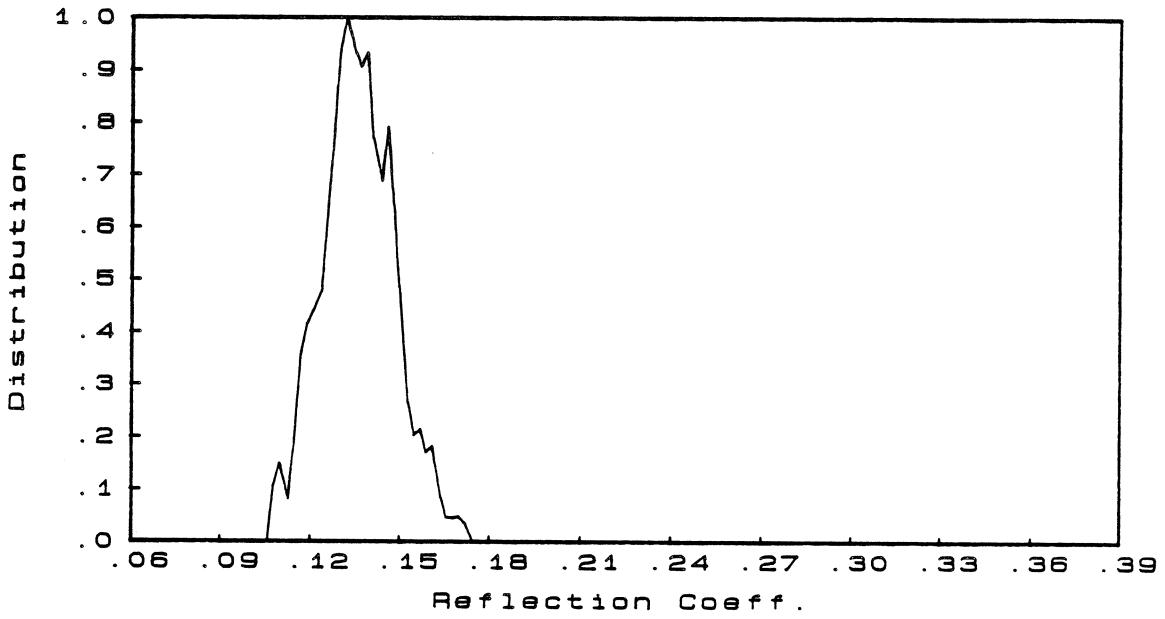
Fig. 19 Monte Carlo analysis of the fixed tolerance design which considered R=50 uniformly distributed outcomes, rather than vertices, and with no specification on either the reflection coefficient or the least squares term of the objective function.



FILE: VAOR50
RANGES OF REFLECTION COEFFICIENTS : .1167 .2583

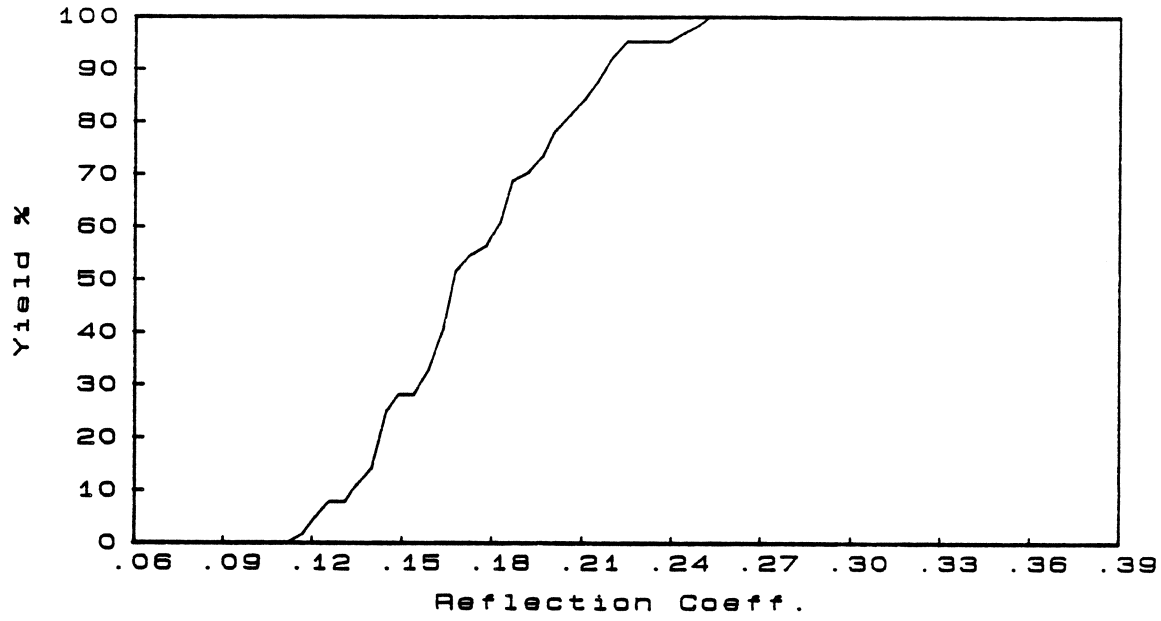
Fig. 20 Vertex analysis of the fixed tolerance design which considered R=50 uniformly distributed outcomes, rather than vertices, and with no specification on either the reflection coefficient or the least squares term of the objective function.

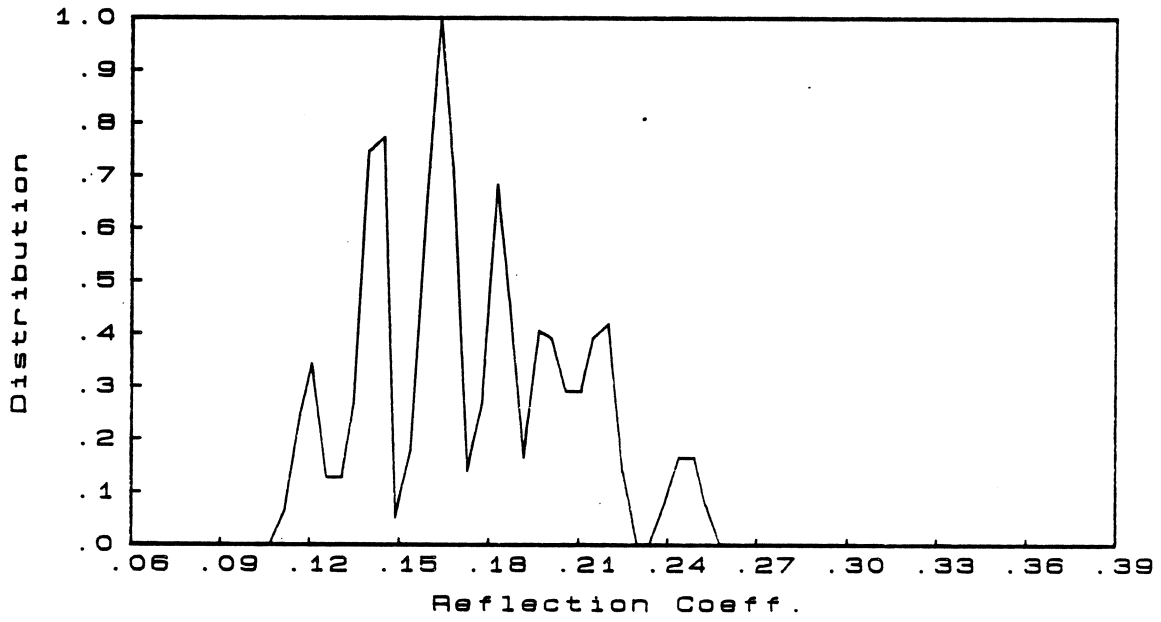




FILE: MC1R50
RANGES OF REFLECTION COEFFICIENTS : .1081 .1724

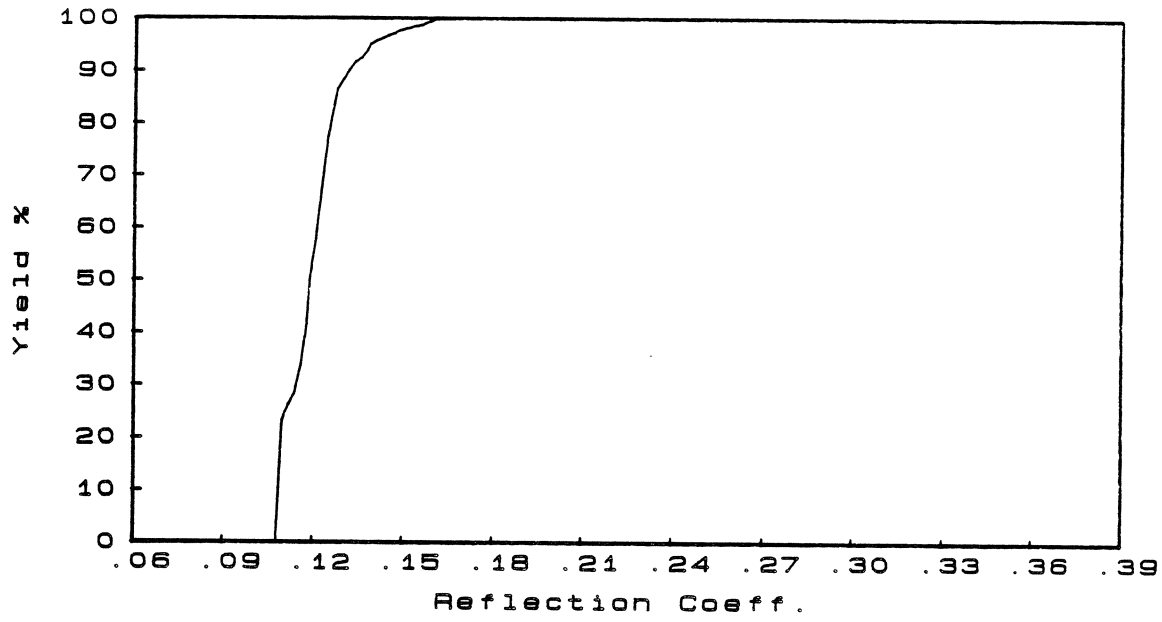
Fig. 21 Monte Carlo analysis of the fixed tolerance design which considered R=50 uniformly distributed outcomes, rather than vertices, and with a specification value of 0.002 imposed on the least squares term of the objective function.

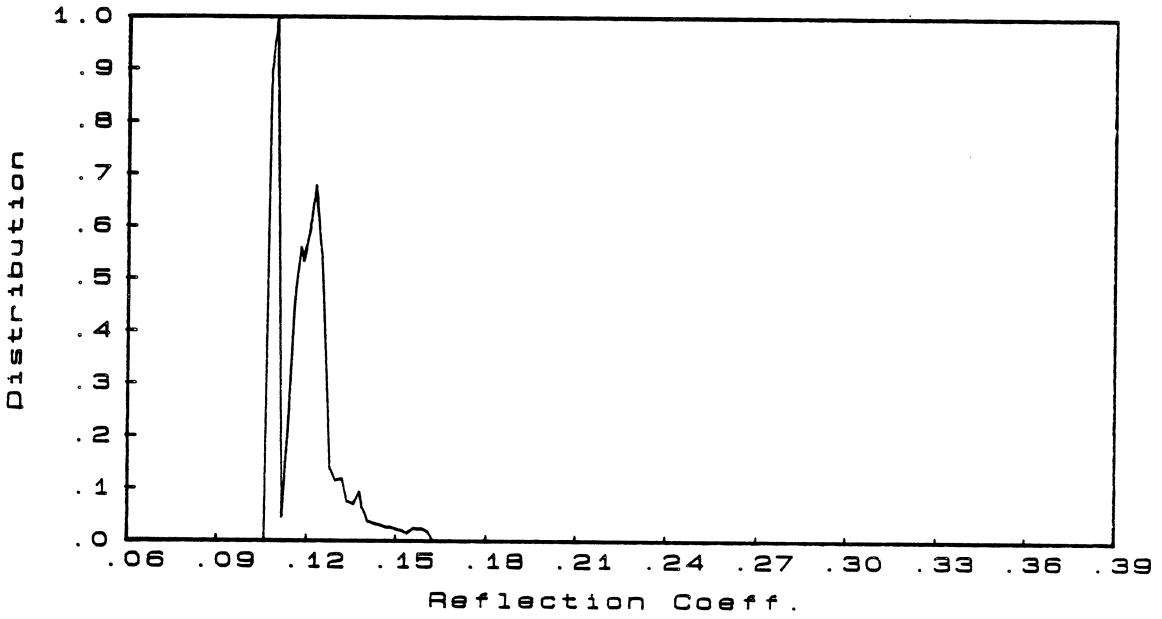




FILE: VA1R50
RANGES OF REFLECTION COEFFICIENTS : .1166 .2532

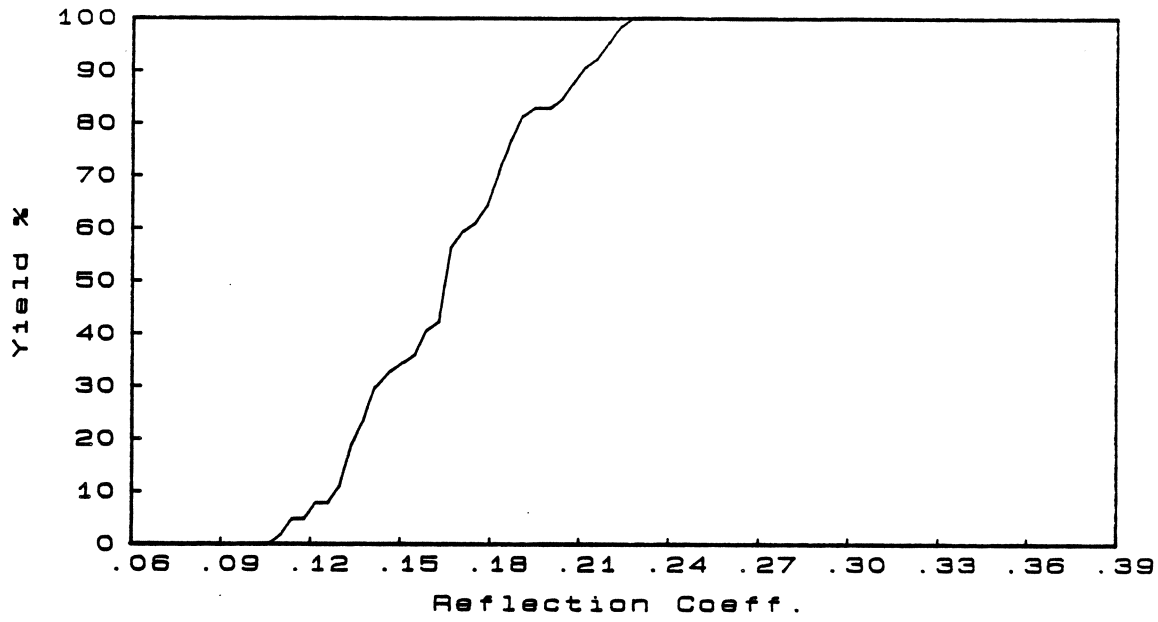
Fig. 22 Vertex analysis of the fixed tolerance design which considered R=50 uniformly distributed outcomes, rather than vertices, and with a specification value of 0.002 imposed on the least squares term of the objective function.

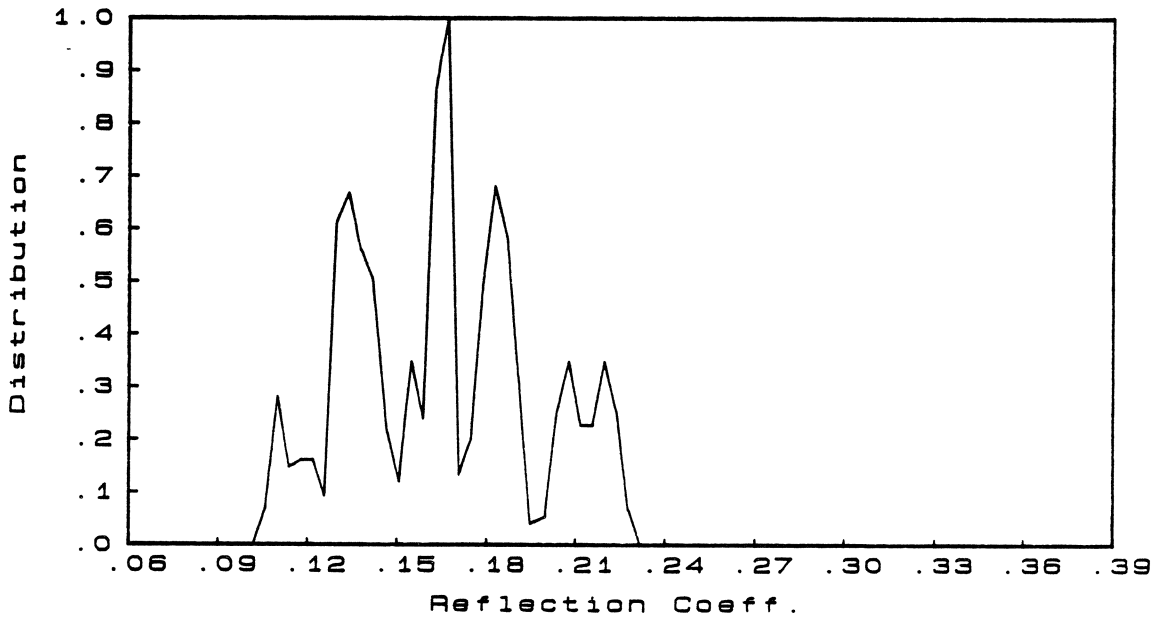




FILE: MC2R50
RANGES OF REFLECTION COEFFICIENTS : .1084 .1613

Fig. 23 Monte Carlo analysis of the fixed tolerance design which considered R=50 uniformly distributed outcomes, rather than vertices, and with a specification value of 0.119 imposed on each of the reflection coefficients term of the objective function.





FILE: VA2R50
RANGES OF REFLECTION COEFFICIENTS : .1097 .2281

Fig. 24 Vertex analysis of the fixed tolerance design which considered R=50 uniformly distributed outcomes, rather than vertices, and with a specification value of 0.119 imposed on each of the reflection coefficients term of the objective function.

DISCUSSION AND SUGGESTIONS

The TOLCAD package in its present form does not allow the user to define general lower and upper specifications on a performance function of interest to formulate a general design problem. All it offers is a fixed set of least squares objective functions for certain response functions or S parameters of the circuit. From the work done up to now it seems that several ideas and suggestions, if implemented, would significantly increase the capabilities of the package in terms of the range and type of problems that could be solved.

One suggestion is to create access to individual responses at user defined frequency points and to store that information in a separate array or arrays. It would also be desirable to create an option for the user to define lower and upper specifications for any performance function of interest at arbitrarily defined frequency points or intervals. This, in conjunction with the access to individual responses, would lead to the possibility of creating error functions at user defined frequencies, which in turn makes possible, for example, the definition of a design problem in terms of minimax optimization. The next logical step would be to incorporate minimax optimization for the nominal design problem as well as for the fixed tolerance problem.

In order to be able to evaluate solutions to the FTP in terms of yield it would be very useful to have the option of performing a Monte Carlo analysis for a given design in a way that would lead directly to the calculation of yield of circuit outcomes. At present the "Monte Carlo" analysis in TOLCAD, being the original COMPACT formulation, appears to be carried out at a single frequency at a time. It is not employed in any of the analyses reported here.

Another suggestion for further development is to create an automatic scheme for reducing the number of frequency points used in optimization retaining at the same time the original frequency points defined by the user for analysis purposes.

The specialized test results provide important information on the potential benefits of further development of TOLCAD. Optimization based on random circuit outcomes appears promising. Optimization involving the weighted nominal point as well as the vertices appears promising also. Reducing execution times by reducing the overall number of response evaluations needed for adequate design centering should be addressed in future studies.

FILE: MCNOM
DATA: .32285 2.89640 .76090 2.28120 .96591 .96759
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1036 .2351

SPECIFICATION	YIELD %	PROB. DENSITY
.0900	.0000	.0000
.0946	.0000	.0000
.0991	.0000	.0000
.1036	.1000	.1044
.1082	3.4000	.4534
.1127	11.4000	.7449
.1172	21.8000	.9451
.1218	34.6000	1.0000
.1263	46.0000	.7693
.1308	53.7000	.4686
.1354	58.2000	.2789
.1399	61.3000	.2353
.1444	64.6000	.3404
.1490	69.3000	.2459
.1535	70.7000	.1236
.1580	72.9000	.1613
.1626	74.7000	.1196
.1671	76.9000	.4329
.1716	84.6000	.4243
.1762	86.6000	.1024
.1807	88.1000	.1038
.1852	89.4000	.1295
.1898	91.3000	.1355
.1943	92.6000	.0687
.1988	93.2000	.0575
.2034	94.2000	.0879
.2079	95.4000	.1058
.2124	96.9000	.1474
.2170	98.8000	.0886
.2215	99.1000	.0245
.2261	99.6000	.0185
.2306	99.6000	.0152
.2351	100.0000	.0185
.2397	100.0000	.0000
.2442	100.0000	.0000
.2487	100.0000	.0000

FILE: VANOM
DATA: .32285 2.89640 .76090 2.28120 .96591 .96759
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1078 .3178

SPECIFICATION	YIELD %	PROB. DENSITY
.0861	.0000	.0000
.0934	.0000	.0000
.1006	.0000	.0606
.1078	1.5625	.4242
.1151	6.2500	.2121
.1223	6.2500	.5606
.1296	15.6250	.7727
.1368	18.7500	.5758
.1440	26.5625	1.0000
.1513	34.3750	.6364
.1585	37.5000	.2424
.1658	40.6250	.9242
.1730	51.5625	.6818
.1803	51.5625	.0909
.1875	54.6875	.3182
.1947	56.2500	.0152
.2020	56.2500	.3788
.2092	62.5000	.5455
.2165	65.6250	.7424
.2237	75.0000	.8939
.2309	79.6875	.4394
.2382	82.8125	.1515
.2454	82.8125	.0758
.2527	84.3750	.0758
.2599	84.3750	.1515
.2672	87.5000	.5303
.2744	92.1875	.2576
.2816	92.1875	.1667
.2889	95.3125	.2121
.2961	95.3125	.0000
.3034	95.3125	.0000
.3106	95.3125	.3182
.3178	100.0000	.3182
.3251	100.0000	.0000
.3323	100.0000	.0000
.3396	100.0000	.0000

FILE: MCTOLA
DATA: .30052 2.67090 .70563 2.11650 .89034 .90129
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1295 .1848

SPECIFICATION	YIELD %	PROB. DENSITY
.1238	.0000	.0000
.1257	.0000	.0000
.1276	.0000	.0011
.1295	.1000	.0415
.1314	.7000	.1106
.1334	1.7000	.1117
.1353	2.4000	.0915
.1372	3.2000	.0979
.1391	4.1000	.1766
.1410	6.1000	.3213
.1429	9.0000	.3968
.1448	12.2000	.4138
.1467	15.6000	.4830
.1486	20.0000	.6840
.1505	26.0000	.6989
.1524	31.1000	.8021
.1543	38.7000	1.0000
.1562	46.2000	.8915
.1581	52.8000	.8851
.1600	60.1000	.8809
.1619	66.7000	.9521
.1638	74.6000	.8191
.1658	79.3000	.5649
.1677	83.9000	.5638
.1696	88.0000	.4670
.1715	91.2000	.3734
.1734	93.8000	.2564
.1753	95.2000	.1138
.1772	95.9000	.1351
.1791	97.3000	.1309
.1810	97.9000	.0926
.1829	98.9000	.1500
.1848	100.0000	.0713
.1867	100.0000	.0000
.1886	100.0000	.0000
.1905	100.0000	.0000

FILE: VATOLA
DATA: .30052 2.67090 .70563 2.11650 .89034 .90129
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1192 .2450

SPECIFICATION	YIELD %	PROB. DENSITY
.1062	.0000	.0000
.1105	.0000	.0000
.1149	.0000	.0909
.1192	1.5625	.2121
.1235	3.1250	.0758
.1279	3.1250	.0303
.1322	4.6875	.5303
.1365	10.9375	.3333
.1409	10.9375	.4242
.1452	18.7500	.8030
.1496	23.4375	.5303
.1539	28.1250	.4545
.1582	31.2500	.2576
.1626	34.3750	1.0000
.1669	46.8750	.9545
.1712	50.0000	.5606
.1756	57.8125	.8333
.1799	64.0625	.9394
.1843	73.4375	.8788
.1886	78.1250	.2879
.1929	79.6875	.3182
.1973	84.3750	.6970
.2016	90.6250	.5606
.2059	93.7500	.3636
.2103	96.8750	.1667
.2146	96.8750	.0758
.2189	98.4375	.1061
.2233	98.4375	.0000
.2276	98.4375	.0000
.2320	98.4375	.0000
.2363	98.4375	.0000
.2406	98.4375	.1061
.2450	100.0000	.1061
.2493	100.0000	.0000
.2536	100.0000	.0000
.2580	100.0000	.0000
.2623	100.0000	.0000

FILE: MCTOLB
DATA: .31805 2.11400 .47396 1.42330 .70526 .95486
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1891 .2362

SPECIFICATION	YIELD %	PROB. DENSITY
.1843	.0000	.0000
.1859	.0000	.0000
.1875	.0000	.0012
.1891	.1000	.0401
.1908	.7000	.1498
.1924	2.2000	.2429
.1940	4.1000	.3137
.1956	6.7000	.4422
.1972	10.1000	.4257
.1989	12.7000	.3844
.2005	16.0000	.6262
.2021	21.3000	.7134
.2037	25.8000	.6262
.2054	30.6000	.8160
.2070	37.3000	1.0000
.2086	44.1000	.8078
.2102	49.0000	.7972
.2118	55.7000	.9976
.2135	62.5000	.8172
.2151	67.5000	.8243
.2167	74.1000	.8172
.2183	78.6000	.5200
.2200	82.0000	.5719
.2216	86.6000	.5554
.2232	89.6000	.3915
.2248	92.3000	.3479
.2264	94.4000	.2252
.2281	95.5000	.1002
.2297	96.1000	.1368
.2313	97.4000	.1392
.2329	98.0000	.0955
.2346	98.9000	.1580
.2362	100.0000	.0802
.2378	100.0000	.0000
.2394	100.0000	.0000
.2410	100.0000	.0000

FILE: VATOLB
DATA: .31805 2.11400 .47396 1.42330 .70526 $.9548_6$
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1763 $.250_0$

SPECIFICATION	YIELD %	PROB. DENSITY
.1686	.0000	.0000
.1712	.0000	.0000
.1737	.0000	.0962
.1763	1.5625	.4038
.1788	4.6875	.2115
.1814	4.6875	.2308
.1839	7.8125	.2308
.1864	7.8125	.1923
.1890	10.9375	.4808
.1915	14.0625	.6346
.1941	18.7500	.3269
.1966	18.7500	.1346
.1992	21.8750	.7692
.2017	28.1250	.6731
.2043	31.2500	.8654
.2068	39.0625	.5385
.2093	39.0625	.5385
.2119	46.8750	.8654
.2144	50.0000	.6731
.2170	56.2500	.6923
.2195	59.3750	.6923
.2221	65.6250	.7308
.2246	68.7500	.4423
.2271	71.8750	.2885
.2297	73.4375	.5577
.2322	79.6875	1.0000
.2348	85.9375	.8462
.2373	90.6250	.4231
.2399	92.1875	.3077
.2424	95.3125	.5192
.2449	98.4375	.2115
.2475	98.4375	.0962
.2500	100.0000	.1346
.2526	100.0000	.0000
.2551	100.0000	.0000
.2577	100.0000	.0000

FILE: MC1N64
DATA: .30217 2.31953 .59604 1.76589 .76941 .89357
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1626 .2146

SPECIFICATION	YIELD %	PROB. DENSITY
.1572	.0000	.0000
.1590	.0000	.0000
.1608	.0000	.0000
.1626	.1000	.0807
.1644	1.2000	.1472
.1661	2.2000	.1418
.1679	3.3000	.0981
.1697	3.9000	.1461
.1715	5.8000	.3348
.1733	8.8000	.4068
.1751	11.8000	.3621
.1769	14.6000	.4580
.1787	19.0000	.6576
.1805	24.4000	.7110
.1823	29.9000	.8004
.1841	36.6000	.8877
.1859	43.2000	.8233
.1877	49.4000	.8844
.1895	56.8000	1.0000
.1913	64.3000	.9095
.1930	70.7000	.8637
.1948	77.2000	.6478
.1966	80.8000	.5376
.1984	85.7000	.5889
.2002	89.4000	.4035
.2020	92.0000	.3130
.2038	94.2000	.2236
.2056	95.4000	.1069
.2074	96.1000	.1330
.2092	97.4000	.1276
.2110	98.0000	.0883
.2128	98.9000	.1461
.2146	100.0000	.0742
.2164	100.0000	.0000
.2181	100.0000	.0000
.2199	100.0000	.0000
.2217	100.0000	.0000

FILE: VA1N64
DATA: .30217 2.31953 .59604 1.76589 .76941 .89357
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1486 .2677

SPECIFICATION	YIELD %	PROB. DENSITY
.1363	.0000	.0000
.1404	.0000	.0000
.1445	.0000	.0789
.1486	1.5625	.1711
.1528	3.1250	.1579
.1569	4.6875	.1711
.1610	6.2500	.0658
.1651	6.2500	.0000
.1692	7.8125	1.0000
.1733	23.4375	.9342
.1774	25.0000	.3684
.1815	32.8125	.7500
.1856	39.0625	.6842
.1897	46.8750	.9342
.1938	56.2500	.6184
.1979	59.3750	.4211
.2020	65.6250	.6711
.2061	71.8750	.5658
.2102	76.5625	.3684
.2144	79.6875	.3947
.2185	84.3750	.4079
.2226	87.5000	.3158
.2267	90.6250	.2368
.2308	92.1875	.1447
.2349	93.7500	.1711
.2390	95.3125	.0526
.2431	95.3125	.1711
.2472	98.4375	.1842
.2513	98.4375	.0000
.2554	98.4375	.0000
.2595	98.4375	.0000
.2636	98.4375	.0921
.2677	100.0000	.0921
.2718	100.0000	.0000
.2759	100.0000	.0000
.2801	100.0000	.0000

FILE: WKWTMC
DATA: .31201 2.80780 .73944 2.21830 .93594 .93601
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1107 .1644

SPECIFICATION	YIELD %	PROB. DENSITY
.1052	.0000	.0000
.1070	.0000	.0000
.1089	.0000	.0000
.1107	.1000	.0824
.1126	1.2000	.1011
.1144	1.7000	.1116
.1163	3.1000	.1898
.1181	4.6000	.1783
.1200	6.2000	.2993
.1218	9.4000	.4369
.1237	12.8000	.3670
.1255	15.5000	.4254
.1274	19.8000	.5923
.1292	24.6000	.5631
.1311	28.8000	.5443
.1329	33.5000	.6434
.1348	38.8000	.5839
.1366	42.9000	.5464
.1385	48.0000	.7508
.1403	54.7000	.8175
.1422	61.2000	1.0000
.1440	70.1000	.8968
.1459	75.3000	.6882
.1477	81.5000	.7122
.1496	86.4000	.5735
.1514	90.6000	.4213
.1533	93.1000	.2471
.1551	94.7000	.1335
.1570	95.5000	.1408
.1588	97.0000	.1648
.1607	98.0000	.1116
.1625	98.9000	.1356
.1644	100.0000	.0709
.1662	100.0000	.0000
.1681	100.0000	.0000
.1699	100.0000	.0000
.1718	100.0000	.0000

FILE: WKWTVA
DATA: .31201 2.80780 .73944 2.21830 .93594 .93601
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1205 .2515

SPECIFICATION	YIELD %	PROB. DENSITY
.1070	.0000	.0000
.1115	.0000	.0000
.1160	.0000	.0725
.1205	1.5625	.2754
.1250	4.6875	.3623
.1296	7.8125	.3623
.1341	10.9375	.2174
.1386	12.5000	.3913
.1431	18.7500	1.0000
.1476	28.1250	.5217
.1521	28.1250	.0580
.1567	31.2500	.5652
.1612	37.5000	.6377
.1657	42.1875	.5217
.1702	46.8750	.4058
.1747	50.0000	.5072
.1792	56.2500	.7536
.1838	62.5000	.5217
.1883	65.6250	.3188
.1928	68.7500	.3188
.1973	71.8750	.5362
.2018	78.1250	.6232
.2063	82.8125	.6232
.2109	89.0625	.5652
.2154	92.1875	.1449
.2199	92.1875	.0000
.2244	92.1875	.2029
.2289	95.3125	.2029
.2334	95.3125	.0000
.2380	95.3125	.1014
.2425	96.8750	.0725
.2470	96.8750	.1884
.2515	100.0000	.2029
.2560	100.0000	.0000
.2605	100.0000	.0000
.2651	100.0000	.0000
.2696	100.0000	.0000

FILE: MCNOM
DATA: .32285 2.89645 .76090 2.28118 .96591 .96759
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1037 .2351

SPECIFICATION	YIELD %	PROB. DENSITY
.0901	.0000	.0000
.0946	.0000	.0000
.0991	.0000	.0000
.1037	.1000	.1037
.1082	3.4000	.4584
.1127	11.5000	.7437
.1173	21.8000	.9445
.1218	34.7000	1.0000
.1263	46.0000	.7635
.1308	53.7000	.4690
.1354	58.2000	.2787
.1399	61.3000	.2351
.1444	64.6000	.3402
.1490	69.3000	.2457
.1535	70.7000	.1235
.1580	72.9000	.1612
.1626	74.7000	.1196
.1671	76.9000	.4326
.1716	84.6000	.4240
.1762	86.6000	.1024
.1807	88.1000	.1037
.1852	89.4000	.1295
.1898	91.3000	.1354
.1943	92.6000	.0687
.1988	93.2000	.0575
.2034	94.2000	.0878
.2079	95.4000	.1057
.2124	96.9000	.1473
.2170	98.8000	.0885
.2215	99.1000	.0244
.2260	99.6000	.0185
.2306	99.6000	.0152
.2351	100.0000	.0185
.2396	100.0000	.0000
.2442	100.0000	.0000
.2487	100.0000	.0000

FILE: VANOM
DATA: .32285 2.89645 .76090 2.28118 .96591 .96759
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1078 .3178

SPECIFICATION	YIELD %	PROB. DENSITY
.0861	.0000	.0000
.0933	.0000	.0000
.1006	.0000	.0606
.1078	1.5625	.4242
.1151	6.2500	.2121
.1223	6.2500	.5606
.1295	15.6250	.7727
.1368	18.7500	.5758
.1440	26.5625	1.0000
.1513	34.3750	.6364
.1585	37.5000	.2424
.1658	40.6250	.9242
.1730	51.5625	.6818
.1802	51.5625	.0909
.1875	54.6875	.3182
.1947	56.2500	.0152
.2020	56.2500	.3788
.2092	62.5000	.5455
.2164	65.6250	.7424
.2237	75.0000	.8939
.2309	79.6875	.4394
.2382	82.8125	.1515
.2454	82.8125	.0758
.2527	84.3750	.0758
.2599	84.3750	.1515
.2671	87.5000	.5303
.2744	92.1875	.2576
.2816	92.1875	.1667
.2889	95.3125	.2121
.2961	95.3125	.0000
.3033	95.3125	.0000
.3106	95.3125	.3182
.3178	100.0000	.3182
.3251	100.0000	.0000
.3323	100.0000	.0000
.3395	100.0000	.0000

FILE: MCON64
DATA: .30052 2.67090 .70564 2.11651 .89035 .90130
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1295 .1848

SPECIFICATION	YIELD %	PROB. DENSITY
.1238	.0000	.0000
.1257	.0000	.0000
.1276	.0000	.0011
.1295	.1000	.0415
.1314	.7000	.1106
.1333	1.7000	.1117
.1353	2.4000	.0915
.1372	3.2000	.0979
.1391	4.1000	.1766
.1410	6.1000	.3213
.1429	9.0000	.3968
.1448	12.2000	.4138
.1467	15.6000	.4830
.1486	20.0000	.6840
.1505	26.0000	.6989
.1524	31.1000	.8021
.1543	38.7000	1.0000
.1562	46.2000	.8915
.1581	52.8000	.8851
.1600	60.1000	.8809
.1619	66.7000	.9521
.1638	74.6000	.8191
.1658	79.3000	.5649
.1677	83.9000	.5638
.1696	88.0000	.4670
.1715	91.2000	.3734
.1734	93.8000	.2564
.1753	95.2000	.1138
.1772	95.9000	.1351
.1791	97.3000	.1309
.1810	97.9000	.0926
.1829	98.9000	.1500
.1848	100.0000	.0713
.1867	100.0000	.0000
.1886	100.0000	.0000
.1905	100.0000	.0000
.1924	100.0000	.0000

FILE: VAON64
DATA: .30052 2.67090 .70564 2.11651 .89035 .90130
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1192 .2450

SPECIFICATION	YIELD %	PROB. DENSITY
.1062	.0000	.0000
.1105	.0000	.0000
.1149	.0000	.0909
.1192	1.5625	.2121
.1235	3.1250	.0758
.1279	3.1250	.0303
.1322	4.6875	.5303
.1366	10.9375	.3333
.1409	10.9375	.4242
.1452	18.7500	.8030
.1496	23.4375	.5303
.1539	28.1250	.4545
.1582	31.2500	.2576
.1626	34.3750	1.0000
.1669	46.8750	.9545
.1712	50.0000	.5606
.1756	57.8125	.8333
.1799	64.0625	.9394
.1843	73.4375	.8636
.1886	78.1250	.4091
.1929	81.2500	.3182
.1973	84.3750	.5758
.2016	90.6250	.5758
.2059	93.7500	.3636
.2103	96.8750	.1667
.2146	96.8750	.0758
.2190	98.4375	.1061
.2233	98.4375	.0000
.2276	98.4375	.0000
.2320	98.4375	.0000
.2363	98.4375	.0000
.2406	98.4375	.1061
.2450	100.0000	.1061
.2493	100.0000	.0000
.2536	100.0000	.0000
.2580	100.0000	.0000
.2623	100.0000	.0000

FILE: MCON20
DATA: .31804 2.11401 .47397 1.42328 .70526 .95486
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1891 .2362

SPECIFICATION	YIELD %	PROB. DENSITY
.1843	.0000	.0000
.1859	.0000	.0000
.1875	.0000	.0012
.1891	.1000	.0401
.1908	.7000	.1498
.1924	2.2000	.2429
.1940	4.1000	.3137
.1956	6.7000	.4422
.1973	10.1000	.4257
.1989	12.7000	.3844
.2005	16.0000	.6262
.2021	21.3000	.7134
.2037	25.8000	.6262
.2054	30.6000	.8160
.2070	37.3000	1.0000
.2086	44.1000	.8078
.2102	49.0000	.7972
.2119	55.7000	.9976
.2135	62.5000	.8172
.2151	67.5000	.8243
.2167	74.1000	.8172
.2183	78.6000	.5200
.2200	82.0000	.5719
.2216	86.6000	.5554
.2232	89.6000	.3915
.2248	92.3000	.3479
.2265	94.4000	.2252
.2281	95.5000	.1002
.2297	96.1000	.1368
.2313	97.4000	.1392
.2329	98.0000	.0955
.2346	98.9000	.1580
.2362	100.0000	.0802
.2378	100.0000	.0000
.2394	100.0000	.0000
.2411	100.0000	.0000
.2427	100.0000	.0000

FILE: VAON20
DATA: .31804 2.11401 .47397 1.42328 .70526 .95486
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1763 .2500

SPECIFICATION	YIELD %	PROB. DENSITY
.1686	.0000	.0000
.1712	.0000	.0000
.1737	.0000	.0962
.1763	1.5625	.4038
.1788	4.6875	.2115
.1814	4.6875	.2308
.1839	7.8125	.2308
.1864	7.8125	.1923
.1890	10.9375	.4808
.1915	14.0625	.6346
.1941	18.7500	.3269
.1966	18.7500	.1346
.1992	21.8750	.7692
.2017	28.1250	.6731
.2042	31.2500	.8654
.2068	39.0625	.5385
.2093	39.0625	.5385
.2119	46.8750	.8654
.2144	50.0000	.6731
.2170	56.2500	.6923
.2195	59.3750	.6923
.2221	65.6250	.7308
.2246	68.7500	.4423
.2271	71.8750	.2885
.2297	73.4375	.5577
.2322	79.6875	1.0000
.2348	85.9375	.8462
.2373	90.6250	.4231
.2399	92.1875	.3077
.2424	95.3125	.5192
.2449	98.4375	.2115
.2475	98.4375	.0962
.2500	100.0000	.1346
.2526	100.0000	.0000
.2551	100.0000	.0000
.2577	100.0000	.0000
.2602	100.0000	.0000

FILE: MC1N64
DATA: .30217 2.31953 .59604 1.76589 .76941 .89357
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1626 .2146

SPECIFICATION	YIELD %	PROB. DENSITY
.1572	.0000	.0000
.1590	.0000	.0000
.1608	.0000	.0000
.1626	.1000	.0807
.1644	1.2000	.1472
.1662	2.2000	.1418
.1679	3.3000	.0981
.1697	3.9000	.1461
.1715	5.8000	.3348
.1733	8.8000	.4068
.1751	11.8000	.3621
.1769	14.6000	.4580
.1787	19.0000	.6576
.1805	24.4000	.7110
.1823	29.9000	.8004
.1841	36.6000	.8877
.1859	43.2000	.8233
.1877	49.4000	.8844
.1895	56.8000	1.0000
.1913	64.3000	.9095
.1930	70.7000	.8637
.1948	77.2000	.6478
.1966	80.8000	.5376
.1984	85.7000	.5889
.2002	89.4000	.4035
.2020	92.0000	.3130
.2038	94.2000	.2236
.2056	95.4000	.1069
.2074	96.1000	.1330
.2092	97.4000	.1276
.2110	98.0000	.0883
.2128	98.9000	.1461
.2146	100.0000	.0742
.2164	100.0000	.0000
.2182	100.0000	.0000
.2199	100.0000	.0000
.2217	100.0000	.0000

FILE: VA1N64
DATA: .30217 2.31953 .59604 1.76589 .76941 .89357
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1487 .2677

SPECIFICATION	YIELD %	PROB. DENSITY
.1363	.0000	.0000
.1404	.0000	.0000
.1445	.0000	.0789
.1487	1.5625	.1711
.1528	3.1250	.1579
.1569	4.6875	.1711
.1610	6.2500	.0658
.1651	6.2500	.0000
.1692	7.8125	1.0000
.1733	23.4375	.9342
.1774	25.0000	.3684
.1815	32.8125	.7500
.1856	39.0625	.6842
.1897	46.8750	.9342
.1938	56.2500	.6184
.1979	59.3750	.4211
.2020	65.6250	.6711
.2061	71.8750	.5658
.2102	76.5625	.3684
.2144	79.6875	.3947
.2185	84.3750	.4079
.2226	87.5000	.3158
.2267	90.6250	.2368
.2308	92.1875	.1447
.2349	93.7500	.1711
.2390	95.3125	.0526
.2431	95.3125	.1711
.2472	98.4375	.1842
.2513	98.4375	.0000
.2554	98.4375	.0000
.2595	98.4375	.0000
.2636	98.4375	.0921
.2677	100.0000	.0921
.2718	100.0000	.0000
.2759	100.0000	.0000
.2801	100.0000	.0000

FILE: MC1N20
DATA: .29946 2.27226 .57255 1.73030 .75552 .90911
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1668 .2187

SPECIFICATION	YIELD %	PROB. DENSITY
.1614	.0000	.0000
.1632	.0000	.0000
.1650	.0000	.0000
.1668	.1000	.0823
.1686	1.2000	.1513
.1704	2.2000	.1368
.1721	3.2000	.0912
.1739	3.8000	.1591
.1757	5.8000	.3437
.1775	8.7000	.3904
.1793	11.5000	.3560
.1811	14.3000	.4605
.1829	18.6000	.6707
.1847	24.1000	.7353
.1865	29.6000	.8053
.1883	36.2000	.9143
.1901	43.0000	.8498
.1918	49.1000	.8865
.1936	56.4000	1.0000
.1954	63.7000	.8932
.1972	69.9000	.8977
.1990	76.8000	.7008
.2008	80.5000	.5606
.2026	85.5000	.5996
.2044	89.1000	.3993
.2062	91.7000	.3437
.2080	94.2000	.2514
.2098	95.4000	.1057
.2115	96.1000	.1368
.2133	97.4000	.1212
.2151	97.9000	.0901
.2169	98.9000	.1580
.2187	100.0000	.0745
.2205	100.0000	.0000
.2223	100.0000	.0000
.2241	100.0000	.0000

FILE: VA1N20
DATA: .29946 2.27226 .57255 1.73030 .75552 .90911
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1538 .2594

SPECIFICATION	YIELD %	PROB. DENSITY
.1429	.0000	.0000
.1465	.0000	.0000
.1502	.0000	.1200
.1538	1.5625	.2800
.1574	3.1250	.0800
.1611	3.1250	.2600
.1647	6.2500	.2000
.1684	6.2500	.4600
.1720	12.5000	.8800
.1757	17.1875	1.0000
.1793	25.0000	.8000
.1829	28.1250	.9800
.1866	37.5000	1.0000
.1902	40.6250	.6400
.1939	46.8750	1.0000
.1975	53.1250	.9800
.2011	59.3750	.8400
.2048	64.0625	.7000
.2084	68.7500	.7400
.2121	73.4375	.5400
.2157	76.5625	.9200
.2194	84.3750	.6200
.2230	84.3750	.1000
.2266	87.5000	.8400
.2303	93.7500	.5200
.2339	93.7500	.0000
.2376	93.7500	.4200
.2412	98.4375	.4200
.2449	98.4375	.0000
.2485	98.4375	.0000
.2521	98.4375	.0000
.2558	98.4375	.1400
.2594	100.0000	.1400
.2631	100.0000	.0000
.2667	100.0000	.0000
.2704	100.0000	.0000

FILE: MC2N64
DATA: .30940 2.66365 .72224 2.16988 .88804 .93191
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1011 .1601

SPECIFICATION	YIELD %	PROB. DENSITY
.0950	.0000	.0000
.0970	.0000	.0000
.0990	.0000	.0011
.1011	.1000	.0411
.1031	.7000	.1095
.1051	1.7000	.1105
.1072	2.4000	.0926
.1092	3.2000	.0842
.1113	3.9000	.1400
.1133	5.7000	.3453
.1153	9.1000	.4137
.1174	12.0000	.3495
.1194	15.0000	.4884
.1214	19.7000	.5916
.1235	24.3000	.6537
.1255	30.2000	.8084
.1275	36.9000	.8621
.1296	43.7000	.8368
.1316	50.4000	.9611
.1336	58.7000	.9600
.1357	65.6000	1.0000
.1377	74.1000	.8253
.1397	78.6000	.6126
.1418	84.3000	.6905
.1438	89.0000	.4611
.1458	91.7000	.3063
.1479	94.0000	.2221
.1499	95.2000	.1000
.1520	95.9000	.1442
.1540	97.4000	.1295
.1560	97.9000	.0832
.1581	98.9000	.1495
.1601	100.0000	.0705
.1621	100.0000	.0000
.1642	100.0000	.0000
.1662	100.0000	.0000
.1682	100.0000	.0000

FILE: VA2N64
DATA: .30940 2.66365 .72224 2.16988 .88804 .93191
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1009 .2231

SPECIFICATION	YIELD %	PROB. DENSITY
.0883	.0000	.0000
.0925	.0000	.0000
.0967	.0000	.0959
.1009	1.5625	.0959
.1051	1.5625	.0000
.1093	1.5625	.0000
.1135	1.5625	.0000
.1177	1.5625	.2603
.1220	6.2500	.4795
.1262	9.3750	.1507
.1304	9.3750	.0000
.1346	9.3750	.1096
.1388	12.5000	.7397
.1430	21.8750	.6849
.1472	25.0000	.4247
.1515	31.2500	.8082
.1557	39.0625	.5479
.1599	42.1875	.5205
.1641	50.0000	1.0000
.1683	59.3750	.6438
.1725	62.5000	.4110
.1767	68.7500	.9315
.1809	78.1250	.5205
.1852	78.1250	.0000
.1894	78.1250	.1233
.1936	81.2500	.6301
.1978	89.0625	.6986
.2020	93.7500	.5068
.2062	98.4375	.2466
.2104	98.4375	.0000
.2147	98.4375	.0000
.2189	98.4375	.0959
.2231	100.0000	.0959
.2273	100.0000	.0000
.2315	100.0000	.0000
.2357	100.0000	.0000
.2399	100.0000	.0000

FILE: MC2N20
DATA: .29673 2.57570 .69975 2.10130 .86106 .89066
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1256 .1828

SPECIFICATION	YIELD %	PROB. DENSITY
.1196	.0000	.0000
.1216	.0000	.0000
.1236	.0000	.0011
.1256	.1000	.0415
.1275	.7000	.1106
.1295	1.7000	.1117
.1315	2.4000	.0936
.1334	3.2000	.0862
.1354	3.9000	.1372
.1374	5.6000	.3234
.1394	8.7000	.3628
.1413	11.2000	.3606
.1433	14.6000	.4819
.1453	18.9000	.6819
.1473	25.0000	.7202
.1492	30.3000	.8319
.1512	38.1000	1.0000
.1532	45.5000	.9160
.1551	52.6000	.9383
.1571	60.2000	.9511
.1591	67.5000	1.0000
.1611	75.3000	.7511
.1630	79.4000	.5819
.1650	84.8000	.6266
.1670	88.8000	.4223
.1690	91.6000	.3404
.1709	94.1000	.2309
.1729	95.2000	.0936
.1749	95.9000	.1298
.1769	97.2000	.1309
.1788	97.9000	.1011
.1808	98.9000	.1489
.1828	100.0000	.0713
.1847	100.0000	.0000
.1867	100.0000	.0000
.1887	100.0000	.0000
.1907	100.0000	.0000

FILE: VA2N20
DATA: .29673 2.57570 .69975 2.10130 .86106 .89066
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1254 .2142

SPECIFICATION	YIELD %	PROB. DENSITY
.1162	.0000	.0000
.1192	.0000	.0000
.1223	.0000	.0921
.1254	1.5625	.0921
.1284	1.5625	.0000
.1315	1.5625	.0000
.1346	1.5625	.0000
.1376	1.5625	.2500
.1407	6.2500	.4079
.1438	9.3750	.4868
.1468	15.6250	.5000
.1499	18.7500	.2895
.1530	21.8750	.3289
.1560	25.0000	.1974
.1591	26.5625	.3421
.1621	32.8125	1.0000
.1652	43.7500	.5395
.1683	43.7500	.1974
.1713	50.0000	.8947
.1744	59.3750	.6053
.1775	62.5000	.5921
.1805	71.8750	.9868
.1836	79.6875	.3684
.1867	79.6875	.0263
.1897	81.2500	.0395
.1928	81.2500	.3421
.1959	87.5000	.4079
.1989	89.0625	.3947
.2020	95.3125	.4474
.2051	96.8750	.0263
.2081	96.8750	.0658
.2112	98.4375	.1842
.2142	100.0000	.0789
.2173	100.0000	.0000
.2204	100.0000	.0000
.2234	100.0000	.0000
.2265	100.0000	.0000

FILE: WMCON64
DATA: .31201 2.80781 .73945 2.21832 .93594 .93601
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1107 .1644

SPECIFICATION	YIELD %	PROB. DENSITY
.1052	.0000	.0000
.1070	.0000	.0000
.1089	.0000	.0000
.1107	.1000	.0824
.1126	1.2000	.1011
.1144	1.7000	.1116
.1163	3.1000	.1898
.1181	4.6000	.1783
.1200	6.2000	.2993
.1218	9.4000	.4369
.1237	12.8000	.3670
.1255	15.5000	.4254
.1274	19.8000	.5923
.1292	24.6000	.5631
.1311	28.8000	.5443
.1329	33.5000	.6434
.1348	38.8000	.5839
.1366	42.9000	.5464
.1385	48.0000	.7508
.1403	54.7000	.8175
.1422	61.2000	1.0000
.1440	70.1000	.8968
.1459	75.3000	.6882
.1477	81.5000	.7122
.1496	86.4000	.5735
.1514	90.6000	.4213
.1533	93.1000	.2471
.1551	94.7000	.1335
.1570	95.5000	.1408
.1588	97.0000	.1648
.1607	98.0000	.1116
.1625	98.9000	.1356
.1644	100.0000	.0709
.1662	100.0000	.0000
.1681	100.0000	.0000
.1699	100.0000	.0000

FILE: WVAON64
DATA: .31201 2.80781 .73945 2.21832 .93594 .93601
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1205 .2515

SPECIFICATION	YIELD %	PROB. DENSITY
.1070	.0000	.0000
.1115	.0000	.0000
.1160	.0000	.0725
.1205	1.5625	.2754
.1250	4.6875	.3623
.1296	7.8125	.3623
.1341	10.9375	.2174
.1386	12.5000	.3913
.1431	18.7500	1.0000
.1476	28.1250	.5217
.1521	28.1250	.0580
.1567	31.2500	.5652
.1612	37.5000	.6377
.1657	42.1875	.5217
.1702	46.8750	.4058
.1747	50.0000	.5072
.1792	56.2500	.7536
.1838	62.5000	.5217
.1883	65.6250	.3188
.1928	68.7500	.3188
.1973	71.8750	.5362
.2018	78.1250	.6232
.2064	82.8125	.6232
.2109	89.0625	.5652
.2154	92.1875	.1449
.2199	92.1875	.0000
.2244	92.1875	.2029
.2289	95.3125	.2029
.2335	95.3125	.0000
.2380	95.3125	.1014
.2425	96.8750	.0725
.2470	96.8750	.1884
.2515	100.0000	.2029
.2560	100.0000	.0000
.2606	100.0000	.0000
.2651	100.0000	.0000
.2696	100.0000	.0000

FILE: WMCON20
DATA: .30174 2.68895 .70850 2.12552 .89631 .90525
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1282 .1834

SPECIFICATION	YIELD %	PROB. DENSITY
.1225	.0000	.0000
.1244	.0000	.0000
.1263	.0000	.0011
.1282	.1000	.0411
.1301	.7000	.1095
.1320	1.7000	.1105
.1339	2.4000	.0895
.1358	3.2000	.1042
.1377	4.2000	.1821
.1396	6.2000	.3158
.1415	9.1000	.4000
.1434	12.4000	.4158
.1453	15.8000	.4853
.1472	20.3000	.6779
.1491	26.2000	.6758
.1510	31.2000	.7863
.1529	38.8000	1.0000
.1548	46.4000	.8747
.1567	52.8000	.8589
.1586	60.1000	.8811
.1605	66.8000	.9516
.1624	74.7000	.7947
.1643	79.2000	.5421
.1662	83.8000	.5747
.1681	88.1000	.4779
.1700	91.3000	.3600
.1719	93.8000	.2463
.1739	95.2000	.1137
.1758	95.9000	.1337
.1777	97.3000	.1295
.1796	97.9000	.0916
.1815	98.9000	.1484
.1834	100.0000	.0705
.1853	100.0000	.0000
.1872	100.0000	.0000
.1891	100.0000	.0000

FILE: WVAON20
DATA: .30174 2.68895 .70850 2.12552 .89631 .90525
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1221 .2472

SPECIFICATION	YIELD %	PROB. DENSITY
.1091	.0000	.0000
.1134	.0000	.0000
.1178	.0000	.0632
.1221	1.5625	.1368
.1264	3.1250	.1158
.1307	4.6875	.1895
.1350	7.8125	.2316
.1393	10.9375	.4737
.1437	18.7500	.4526
.1480	21.8750	.3895
.1523	28.1250	.2737
.1566	28.1250	.0000
.1609	28.1250	.8632
.1652	46.8750	1.0000
.1695	50.0000	.1789
.1739	54.6875	.6000
.1782	64.0625	.6526
.1825	70.3125	.6000
.1868	78.1250	.3053
.1911	78.1250	.0526
.1954	81.2500	.4000
.1998	87.5000	.5684
.2041	93.7500	.2526
.2084	93.7500	.0000
.2127	93.7500	.1368
.2170	96.8750	.2211
.2213	98.4375	.0526
.2256	98.4375	.0000
.2300	98.4375	.0000
.2343	98.4375	.0000
.2386	98.4375	.0000
.2429	98.4375	.0737
.2472	100.0000	.0737
.2515	100.0000	.0000
.2559	100.0000	.0000
.2602	100.0000	.0000

FILE: MCOR50
DATA: .31179 2.79821 .73609 2.22529 .93899 .94455
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1082 .1724

SPECIFICATION	YIELD %	PROB. DENSITY
.1015	.0000	.0000
.1037	.0000	.0000
.1060	.0000	.0000
.1082	.1000	.1082
.1104	1.5000	.1512
.1126	2.2000	.0839
.1148	3.1000	.1976
.1170	5.3000	.3642
.1193	8.4000	.4216
.1215	11.6000	.4547
.1237	15.3000	.4945
.1259	19.3000	.6545
.1281	25.2000	.8124
.1303	31.5000	.9459
.1326	39.3000	1.0000
.1348	46.4000	.9592
.1370	53.8000	.9415
.1392	60.6000	.9327
.1414	67.6000	.7605
.1436	72.3000	.7053
.1458	78.5000	.7947
.1481	83.9000	.6634
.1503	88.4000	.5033
.1525	91.4000	.2781
.1547	92.9000	.2086
.1569	94.7000	.1987
.1591	95.9000	.1766
.1614	97.4000	.1843
.1636	98.5000	.0861
.1658	98.8000	.0475
.1680	99.3000	.0453
.1702	99.5000	.0486
.1724	100.0000	.0364
.1746	100.0000	.0000
.1769	100.0000	.0000
.1791	100.0000	.0000
.1813	100.0000	.0000

FILE: VA0R50
DATA: .31179 2.79821 .73609 2.22529 .93899 .94455
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1167 .2533

SPECIFICATION	YIELD %	PROB. DENSITY
.1026	.0000	.0000
.1073	.0000	.0000
.1120	.0000	.0794
.1167	1.5625	.3016
.1214	4.6875	.4286
.1261	7.8125	.1587
.1308	7.8125	.1587
.1355	10.9375	.3333
.1402	14.0625	.9365
.1450	25.0000	.9683
.1497	28.1250	.0317
.1544	28.1250	.4762
.1591	35.9375	.7778
.1638	40.6250	1.0000
.1685	51.5625	.9206
.1732	54.6875	.3016
.1779	57.8125	.3333
.1826	60.9375	.7302
.1873	68.7500	.6032
.1921	70.3125	.2063
.1968	73.4375	.5079
.2015	78.1250	.4762
.2062	81.2500	.4921
.2109	85.9375	.3651
.2156	87.5000	.3651
.2203	92.1875	.5397
.2250	95.3125	.1746
.2297	95.3125	.0000
.2345	95.3125	.0000
.2392	95.3125	.0952
.2439	96.8750	.2063
.2486	98.4375	.2063
.2533	100.0000	.0952
.2580	100.0000	.0000
.2627	100.0000	.0000
.2674	100.0000	.0000
.2721	100.0000	.0000

FILE: MC1R50
DATA: .31181 2.79805 .73610 2.22511 .93897 .94448
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1081 .1724

SPECIFICATION	YIELD %	PROB. DENSITY
.1015	.0000	.0000
.1037	.0000	.0000
.1059	.0000	.0000
.1081	.1000	.1063
.1104	1.5000	.1486
.1126	2.2000	.0824
.1148	3.1000	.1941
.1170	5.3000	.3579
.1192	8.4000	.4154
.1214	11.6000	.4393
.1237	15.2000	.4794
.1259	19.2000	.6377
.1281	25.0000	.7809
.1303	31.2000	.9382
.1325	39.2000	1.0000
.1347	46.3000	.9338
.1369	53.6000	.9067
.1392	60.3000	.9328
.1414	67.6000	.7711
.1436	72.3000	.6887
.1458	78.5000	.7907
.1480	84.0000	.6432
.1502	88.3000	.4881
.1525	91.4000	.2657
.1547	92.7000	.2039
.1569	94.7000	.2126
.1591	95.9000	.1714
.1613	97.4000	.1811
.1635	98.5000	.0846
.1657	98.8000	.0466
.1680	99.3000	.0445
.1702	99.5000	.0477
.1724	100.0000	.0358
.1746	100.0000	.0000
.1768	100.0000	.0000
.1790	100.0000	.0000
.1812	100.0000	.0000

FILE: VA1R50
DATA: .31181 2.79805 .73610 2.22511 .93897 .94448
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1166 .2532

SPECIFICATION	YIELD %	PROB. DENSITY
.1024	.0000	.0000
.1071	.0000	.0000
.1118	.0000	.0633
.1166	1.5625	.2405
.1213	4.6875	.3418
.1260	7.8125	.1266
.1307	7.8125	.1266
.1354	10.9375	.2658
.1401	14.0625	.7468
.1448	25.0000	.7722
.1495	28.1250	.0506
.1542	28.1250	.1772
.1590	32.8125	.6203
.1637	40.6250	1.0000
.1684	51.5625	.7215
.1731	54.6875	.1392
.1778	56.2500	.2658
.1825	60.9375	.6835
.1872	68.7500	.4684
.1919	70.3125	.1646
.1967	73.4375	.4051
.2014	78.1250	.3924
.2061	81.2500	.2911
.2108	84.3750	.2911
.2155	87.5000	.3924
.2202	92.1875	.4177
.2249	95.3125	.1392
.2296	95.3125	.0000
.2344	95.3125	.0000
.2391	95.3125	.0759
.2438	96.8750	.1646
.2485	98.4375	.1646
.2532	100.0000	.0759
.2579	100.0000	.0000
.2626	100.0000	.0000
.2673	100.0000	.0000
.2721	100.0000	.0000

FILE: MC2R50
DATA: .31483 2.66137 .72780 2.21505 .89042 .97507
NUMBER OF RANDOM OUTCOMES : 1000

RANGES OF REFLECTION COEFFICIENTS : .1084 .1613

SPECIFICATION	YIELD %	PROB. DENSITY
.1029	.0000	.0000
.1047	.0000	.0000
.1065	.0000	.0000
.1084	.1000	.8836
.1102	23.2000	1.0000
.1120	26.2000	.0455
.1138	28.4000	.2217
.1157	33.5000	.4545
.1175	41.6000	.5593
.1193	49.8000	.5338
.1211	57.9000	.5909
.1230	67.6000	.6774
.1248	77.4000	.5510
.1266	83.6000	.2888
.1284	86.5000	.1380
.1303	88.3000	.1147
.1321	90.1000	.1175
.1339	91.7000	.0754
.1357	92.5000	.0698
.1375	93.9000	.0926
.1394	95.1000	.0632
.1412	95.8000	.0371
.1430	96.3000	.0327
.1448	96.8000	.0299
.1467	97.2000	.0261
.1485	97.6000	.0272
.1503	98.0000	.0233
.1521	98.3000	.0200
.1540	98.6000	.0150
.1558	98.8000	.0244
.1576	99.3000	.0233
.1594	99.5000	.0244
.1613	100.0000	.0183
.1631	100.0000	.0000
.1649	100.0000	.0000
.1667	100.0000	.0000

FILE: VA2R50
DATA: .31483 2.66137 .72780 2.21505 .89042 .97507
NUMBER OF VERTICES : 64

RANGES OF REFLECTION COEFFICIENTS : .1097 .2281

SPECIFICATION	YIELD %	PROB. DENSITY
.0975	.0000	.0000
.1016	.0000	.0000
.1056	.0000	.0667
.1097	1.5625	.2800
.1138	4.6875	.1467
.1179	4.6875	.1600
.1220	7.8125	.1600
.1261	7.8125	.0933
.1301	10.9375	.6133
.1342	18.7500	.6667
.1383	23.4375	.5600
.1424	29.6875	.5067
.1465	32.8125	.2133
.1506	34.3750	.1200
.1546	35.9375	.3467
.1587	40.6250	.2400
.1628	42.1875	.8667
.1669	56.2500	1.0000
.1710	59.3750	.1333
.1751	60.9375	.2000
.1791	64.0625	.4933
.1832	70.3125	.6800
.1873	76.5625	.5867
.1914	81.2500	.3200
.1955	82.8125	.0400
.1996	82.8125	.0533
.2036	84.3750	.2533
.2077	87.5000	.3467
.2118	90.6250	.2267
.2159	92.1875	.2267
.2200	95.3125	.3467
.2241	98.4375	.2533
.2281	100.0000	.0667
.2322	100.0000	.0000
.2363	100.0000	.0000
.2404	100.0000	.0000
.2445	100.0000	.0000

SUBROUTINES OF THE TOLCAD PACKAGE

Only subroutines modified or new to Raytheon are listed in full in this report. However, for completeness, all subroutines of COMPSHT are named and their current status indicated as follows.

PROGRAM TOLCAD	MODIFIED	LISTING-3
SUBROUTINE ABCDM	NOT ALTERED	
SUBROUTINE PIMP	NOT ALTERED	
SUBROUTINE SIMP	NOT ALTERED	
SUBROUTINE RPOLAR	NOT ALTERED	
SUBROUTINE MODSIN	NOT ALTERED	
SUBROUTINE COPY	NOT ALTERED	
SUBROUTINE PFUR	NOT ALTERED	
SUBROUTINE TAKEIN	NOT ALTERED	
SUBROUTINE PRGINT	MODIFIED	LISTING-8
SUBROUTINE CASCAD	NOT ALTERED	
SUBROUTINE ANGLE	NOT ALTERED	
SUBROUTINE EEFF1	NOT ALTERED	
SUBROUTINE EEFF2	NOT ALTERED	
SUBROUTINE ZOFRQ2	NOT ALTERED	
SUBROUTINE ANLDAT	NOT ALTERED	
SUBROUTINE ADUMP	NOT ALTERED	
SUBROUTINE SEARCH	MODIFIED	LISTING-10
SUBROUTINE ANALYZ	NOT ALTERED	
SUBROUTINE SNOISE	NOT ALTERED	
SUBROUTINE NODAL	NOT ALTERED	
SUBROUTINE STRIP	NOT ALTERED	
SUBROUTINE WIDTH	NOT ALTERED	

SUBROUTINE SWRIT	NOT ALTERED	
SUBROUTINE ANAERR	NOT ALTERED	
SUBROUTINE SNOIS1	NOT ALTERED	
SUBROUTINE SNOIS2	NOT ALTERED	
SUBROUTINE GRID	NOT ALTERED	
SUBROUTINE FTP	NEW	LISTING-18
SUBROUTINE FTPAQ	NEW	LISTING-20
SUBROUTINE FTPBQ	NEW	LISTING-22
SUBROUTINE FTPCQ	NEW	LISTING-26
SUBROUTINE FTPDQ	NEW	LISTING-29
SUBROUTINE MODEL4	[1]	LISTING-31
SUBROUTINE QPE	[1]	LISTING-34
SUBROUTINE VERT	[1]	LISTING-37
SUBROUTINE ACFUNS	NEW	LISTING-38

Reference

[1] J.W. Bandler and H.L. Abdel-Malek,"Optimal centering,tolerancing and yield determination via updated approximations and cuts," IEEE Trans. Circuits and Systems, vol. CAS-25, 1978, pp.853-871.

```
C     PROGRAM "COMPSHT"
C     THIS IS AN ABBREVIATED VERSION OF CAD4B PASSED THROUGH "CLEAN77".
C     IT CONTAINS BANDLERS VERSION OF SUBROUTINE SEARCH.
      PROGRAM TOLCAD (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE9,TAPE98
     *   ,TAPE99,TAPE15=512,TAPE16=512)
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,W,FR,NT,
     *   NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON /COMB/ U11,U22,ZIN,ATTN,CAB,IND,I,IJ,IJJ,IND3R,INDTR,J,KIMP
     *   ,M,N,NBAND,NCB,NEWEX,NFOPT,PPRD
      COMMON /HVPLOT/ HORI(7),VERT(7),EPLOT(7,7),SENDE(15),BBB(30,2,2)
      COMMON /ANALQ/ ERRD,AOPT,INRJX,JSTK,JSOK,KLM,KYM,MODIM,MODEL,NOOPT
     *   ,IPRI,ITHR,IN1,IN2,IN3,NOIT,IDIV,ITER,IPRAM,ERSET,ERRAN,ERNO,
     *   SAVEC,ILTGT(3),S3P(3,6),VAX(15),IP1(15),IP2(15)
      DOUBLE PRECISION ERRD
      COMMON /DATAS/ PERT,PI,DPR,TWOPI,RPD
      COMMON /ANALZ/ IBAND(3),RIND(70),INC1(70),INC2(70),INC3(70),KX(15)
     *   ,KY(15)
      COMMON /TAKE/ ARRAY(28),IARRAY(28)
      COMMON /TAKE1/ IBUF(136),LDC
      COMMON /TAKE2/ PERUP(3,15)
CBOB  COMMON /CO2/ SEND(15),GR(15),ERR,NOPT,CONV,INEXP
      COMMON /CO2/ SEND(15),GR(15),ERR,NOPT,CONV,INEXP,IGR(15)          12-16-85
      COMMON /CO3/ CARAY(30,2,2),DATX(70,6)
      COMMON /CO4/ VIMP(30,2),FRST(45),DC0(100),DC1(70),DC2(70),DC3(70),
     *   FIG(3,10,5),FIN(3,10),OUT(2,4),WI(3,6),S21(3),FIGG(2,10,5)
      COMMON /CO5/ SLIMP(30,4)
      COMMON /CO6/ S3R(16,3,3)
      COMMON /CO7/ YMAT(15,2,2)
      COMMON /CO8/ PS(30,2,4)
      COMMON /CO9/ AAA(30,2,2),A(2,2),AB1(2,2),AB2(2,2),AB3(2,2),AB4(2,2
     *   )
      COMMON /KSIO/ KSIN,KSOUT
      COMMON /CSERCH/ ALAMDA,G0(15),G1T(15),GA(15),H(15,15),KOUNT,LIMIT,
     *   MAT,NITER,V0,V1T,V01,V02,V03,X0(15),X1(15),X1T(15),XA(15),XST1(
     *   15),XST2(15),LDSERC,NLIST,ILIST
      COMMON /MODIFY/ FN
      COMMON /CNAERR/ IFTLER,AMET,ALOSS
      COMMON /PPLOT/ NPLOT,PARAY(30,2,5),NPOINT,IPNOPR,MIXPR,OLDT
      COMMON /DATANA/ DUMANA(200)
      COMMON /LAY10/ ISTOP,KOPT,IBYP
      COMMON /IOS/ IOT,INPET,INP,IFR,ITO,IED,IOPT,ISCR
      COMMON /BOMB/ JERRX
      COMMON /BUGS/ ABUGS(100),IABFLG
      COMMON /CNOISE/ DX(64),REFR,GN,RN,IN22,INOIS,PAV(10),IA
      COMMON /CGRID/ WRST(15),IFUNCT,IFUX,ICOUNT,IIII,CC,XPER
      COMMON /IVG/ IVG
      COMMON /NFILE/ IPINT
      COMMON /DPLOT/ NPTY
      COMMON /PRTOPT/ NSTAR
      COMMON /CODLAY/ DELAY,FPERT
      COMMON /IMPONE/ NIMP
C
C     INSERT NEW COMMON BLOCK FOR OPTIMIZATION W.R.T.                   12-16-85
C
C#     COMMON/TOLER/NVERT,TOL,NRANK,NCOUNT
```

```
      COMMON/TOLER/NVERT,TOL(15),NRANK,NCOUNT,NCALL,TRES,WIT(65)
C+
      COMMON/QUAMOD/IQUAD,IFTP,IAPPR,MAJOR
      COMMON/MIM/B(136,1),IV(6,64)
      COMMON/LAST/XNOMLST(15),NRFLST(64),FERRLST(64),ZTLST(64,6)
C+
      COMPLEX CARAY
      COMPLEX AAA,A,AB1,AB2,AB3,AB4,BBB,S3R,U11,U22,YMAT,ZIN
      DIMENSION REMAINX(4)
      DOUBLE PRECISION ERRD1,ERRSD,PERT
      EQUIVALENCE (AR1,ARRAY(1))
      DATA REMAINX / "+0600",3*"+0000"/
      DATA AY,AYES,AN,ANO,REP,SAV / "Y","YES","N","NO","REP","SAV"/
C
C *****BEGIN PROGRAM
C
      IBYP = 0
      DPR = 57.29578
      PERT = 1.0D-7
      PI = 3.14159
      RPD = 1.7453292E-2
      TWOPI = 6.28318
C#    TOL=0.05
C#    NRANK=20
C#    NCOUNT=0
      GO TO 30
   10 CONTINUE
      CALL ANAERR
   20 CONTINUE
      IMA = 0
      IMA1 = 0
      WRITE (IOT,310)
      REWIND IFR
      REWIND ITO
   30 IOT = 6
      INPET = 5
      IFR = 15
      ITO = 16
      ISCR = 9
      INP = 15
      IED = 16
      IOPT = 16
C
C***** INITIALIZE SYSTEM SWITCHES
C
   40 NPLOT = 0
      NPOINT = 0
      IBANKF = 0
      IMA = 0
      IMA1 = 0
      ISENS = 0
C
C***** INITIALIZE PROGRAM FILES
C
      CALL PRGINT
```

```
      IF (ISTOP.EQ.1) GO TO 300
C
C***** DETERMINE LINK TO BE CALLED
C
      IF (KOPT.EQ.44) GO TO 60
   50 IF (KOPT.EQ.13) GO TO 140
      IF (KOPT.EQ.27) GO TO 110
C#    GO TO (90,130,150,70,80,100,120), KOPT
      GO TO (90,130,150,70,80,100,120,145), KOPT
   60 IMA = 1
   70 KODE = 5
      GO TO 260
   80 KODE = 6
      GO TO 260
   90 KODE = 9
      GO TO 260
  100 KODE = 8
      GO TO 260
  110 MKODE = 3
  120 KODE = 10
      GO TO 260
  130 ISENS = 1
  140 MKODE = 1
C+
  145 NCOUNT=0
      NCALL=1
C+
  150 LDSERC = 0
      KODE = 1
  160 IF (KODE.NE.1) GO TO 170
      CALL ANLDAT
C#    WRITE(6,*)(DATX(KX(I),KY(I)),I=1,NOPT)
      IF (JERRX.NE.0) GO TO 20
      REWIND IFR
      REWIND ITO
      IF (IBANKF.EQ.0) GO TO 170
      IHOL = IFR
      IFR = ITO
      ITO = IHOL
      REWIND IFR
      REWIND ITO
      CALL ADUMP
      CALL PFUR (REP,IFR,FN,0,ISTA)
C 170 CALL ANALYZ
C
C.    INSERT CALL TO NEW SUBROUTINE "FTPA" FOR OPT. W.R.T.
C
C#    WRITE(6,*)(DATX(KX(I),KY(I)),I=1,NOPT)
C#170 CALL FTPB
C+
  170 IF(KOPT.NE.8)GO TO 175
      IF(NOPT.LE.6) GO TO 172
      PRINT(6,800)NOPT
  800 FORMAT(1X," YOU HAVE SELECTED NOPT=",I4,1X,"VARIABLES."
     +/" THE MAXIMUM ALLOWED CURRENTLY IS 6. PLEASE, CHECK YOUR"
```

```
     +/"  DATA FILE AND TRY AGAIN. PROGRAM TERMINATES HERE.")
      GO TO 300
  172 CALL FTP
      GO TO 176
  175 CALL ANALYZ
  176 CONTINUE
C+
      IF (JERRX.EQ.0) GO TO 180
      IF (IFTLER.NE.23) GO TO 10
      JERRX = 0
      IFTLER = 0
      KODE = 1
      LDSERC = 0
      WRITE (IOT,320) IXX
      GO TO 170
  180 CONTINUE
      IF (KOPT.NE.13) GO TO 190
      IF (NPLOT.LE.0.OR.KODE.NE.1) GO TO 190
  190 CALL SEARCH
      IF (JERRX.NE.0) GO TO 20
      IF (KODE.EQ.2.AND.ISENS.EQ.2) GO TO 290
      IF (LDSERC.GT.0) GO TO 160
      IF (LDSERC.EQ.0) GO TO 210
      IF (IPRIN.EQ.1.OR.MKODE.EQ.1) GO TO 200
      CALL ANLDAT
  200 CALL ANALYZ
  210 IF (KOPT.EQ.2) GO TO 280
  220 PRINT *,"SAVE OPTIMIZED RESULTS (Y/N)     ",
      CALL TAKEIN (INPET,NOITEM)
      IF (ARRAY(1).EQ.AN) GO TO 280
      IF (ARRAY(1).NE.AY) GO TO 220
  230 PRINT *,"OPTIMIZED FILE NAME      ",
      CALL TAKEIN (INPET,NOITEM)
      IF (NOITEM.LE.0) GO TO 230
      IF (IARRAY(1).LT.1) GO TO 230
      IF (IARRAY(1).GT.7) WRITE (IOT,330)
      IF (IARRAY(1).GT.7) GO TO 230
      OPTFIL = AR1
      WRITE (IT0) IOT
      REWIND IT0
      CALL PFUR (SAV,IT0,OPTFIL,0,ISTAT)
      IF (ISTAT.EQ.0) GO TO 250
  240 PRINT *,"FILE IS PERMANENT - REPLACE IT (Y/N)?    "
      CALL TAKEIN (INPET,NOITEM)
      IF (ARRAY(1).EQ.AN.OR.ARRAY(1).EQ.ANO) GO TO 230
      IF (ARRAY(1).NE.AY.AND.ARRAY(1).NE.AYES) GO TO 240
  250 CALL ADUMP
      CALL PFUR (REP,IFR,OPTFIL,0,ISTAT)
      GO TO 280
  260 CALL ANLDAT
      IF (JERRX.NE.0) GO TO 20
      REWIND IFR
      REWIND IT0
      IF (IBANKF.EQ.0) GO TO 270
      IHOL = IFR
```

```
      IFR = ITO
      ITO = IHOL
      REWIND IFR
      REWIND ITO
      CALL ADUMP
      CALL PFUR (REP,IFR,FN,0,ISTA)
  270 CALL ANALYZ
      IF (JERRX.EQ.1) GO TO 10
      IF (IMA.EQ.0) GO TO 280
      IF (KODE.EQ.0) GO TO 280
C
C  **FOLLOW-ON TO COARSE SEARCH**
C
      IMA = 0
      IMA1 = 1
      KOPT = KODE
      GO TO 50
  280 IF (NPLOT.LE.0) GO TO 290
      NPLOT = INC3(NPLOT)
  290 IF (IMA1.EQ.1) IMA = 0
      GO TO 40
  300 CONTINUE
      STOP
C
  310 FORMAT (" EXECUTION STOPPED DUE TO FILE ERROR")
  320 FORMAT (/" EVEN- AND ODD-MODE IMPEDANCES RESET IN LINE",I3,
     *   ", BECAUSE OPTIMIZATION"/" FORCED ZOO>ZOE -- OPTIMIZATION",
     *   " RESTARTS USING NEW VALUES OF ZOE AND ZOO"/" AND",
     *   " CURRENT VALUES OF OTHER VARIABLES, IF ANY"/)
  330 FORMAT (" FILENAME TOO LONG")
C
      END
```

```
      SUBROUTINE PRGINT
C
C *****INITIALIZE THE RUN OR RERUN
C
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,W,FR,NT,
     *   NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON /BOMB/ JERRX
      COMMON /BUGS/ ABUGS(100),IABFLG
      COMMON /IOS/ IOT,INPET,INP,IFR,ITO,IED,IOPT,ISCR
      COMMON /LAY10/ ISTOP,KOPT,IBYP
      COMMON /MODIFY/ FN
      COMMON /TAKE/ ARRAY(28),IARRAY(28)
      COMMON /TAKE1/ IBUF(136),IN2
      DIMENSION AR(2)
      EQUIVALENCE (AR1,AR2,ARRAY(1))
      EQUIVALENCE (AR3,AR(1))
      DATA AP,BL1,BW,RET,XNEW,XNO,YES / "A"," ","BW","RET","NEWS","N",
     *   "Y"/
      DATA QUI,QUI1,INOCH / "QUIT","QUI",10 /
C
C *****SET PROGRAM VARIABLES, WRITE INTRO MESSAGES
C *****BYPASS AFTER FIRST RUN
C
      IF (IBYP.EQ.1) GO TO 10
      IBYP = 1
      ISTOP = 0
      KODE = 0
      JERRX = 0
      KOPT = 0
      FN = BL
      MKODE = 0
      IPRIN = 0
      ITRAP = 0
      IABFLG = 0
      GO TO 20
C
C *****REINITIALIZE PROGRAM VARIABLES FOR SUBSEQUENT PASSES
C
   10 ISTOP = 0
      FN = BL
      KOPT = 0
      MKODE = 0
      KODE = 0
      IPRIN = 0
      JERRX = 0
      ITRAP = 0
      IABFLG = 0
      REWIND 99
C
C *****WHICH OPTION - FILE TO RUN OR QUIT?
C
   20 PRINT *," "
      PRINT *,"'FILE NAME' OR 'QUIT'     ",
      CALL TAKEIN (INPET,NOITEM)
      ILOC = 1
```

```
      JLOC = 1
      IF (AR1.EQ.QUI.OR.AR1.EQ.QUI1) GO TO 80
      AR3 = AR2
      IF (IARRAY(1).LE.INOCH) AR(2) = IBL
      FN = AR3
      CALL PFUR (RET,INP,FN,0,ISTA)
      IF (ISTA.EQ.0) GO TO 30
      WRITE (IOT,100)
      GO TO 20
C
C *****WHICH FUNCTION - 1, 2, 3, 4, 5, 6, 7, 13, 44
C *****SET EXPERT MODE IF THERE ARE/WERE MULTIPLE INPUTS
C
   30 IF (NOITEM.LE.ILOC) GO TO 40
      ILOC = ILOC+1
      IF (IARRAY(JLOC).GT.INOCH) ILOC = ILOC+1
      IPRIN = 1
      GO TO 50
C# 40 PRINT *,"AN(1),SEN(2),OPT(3),SW(4),MAP(5),VAR(6),MC(7),ANOPT(13),R
C#   *ND((44)     ",
   40 PRINT *,"AN(1),SEN(2),OPT(3),SW(4),MAP(5),VAR(6),MC(7),TOL(8),ANOP
     +T(13),RND(44)   ",
      CALL TAKEIN (INPET,NOITEM)
      ILOC = 1
   50 KOPT = ARRAY(ILOC)
C#    IF (KOPT.GE.1.AND.KOPT.LE.7) GO TO 60
      IF (KOPT.GE.1.AND.KOPT.LE.8) GO TO 60
      IF (KOPT.EQ.13) GO TO 60
      IF (KOPT.NE.44) GO TO 40
C
C *****TEST FOR LINE SEQUENCE NUMBERS AND EMPTY FILE
C
   60 REWIND INP
      CALL TAKEIN (INP,NOITEM)
      IF (NOITEM.GT.0) GO TO 70
      WRITE (IOT,110)
      REWIND INP
      GO TO 20
   70 IF (IARRAY(1).EQ.0) ITRAP = 1
      REWIND INP
C
C *****TRANSFER THE FILE TO SCRATCH STORAGE
C
      CALL COPY (INP,IFR)
      GO TO 90
C
C *****QUIT
C
   80 ISTOP = 1
   90 RETURN
  100 FORMAT (" FILE NOT FOUND")
  110 FORMAT (" FILE IS EMPTY")
      END
```

```
      SUBROUTINE SEARCH
C
C        RE:     09/24/85
C
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,
     +W,FR,NT,NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON/TAKE/ ARRAY(28),IARRAY(28)
CBOB  COMMON/CO2/U(15),G(15),V,N,CONV,INEXP
      COMMON/CO2/U(15),G(15),V,N,CONV,INEXP,IGR(15)                    12-18-85
      COMMON/CSERCH/ALAMDA,G0(15),G1T(15),GA(15),H(15,15)
     +         ,KOUNT,LIMIT,MAT,NITER,V0,V1T,V01,V02,V03
     +         ,X0(15),X1(15),X1T(15),XA(15),XST1(15),XST2(15)
     +         ,LDSERC,NLIST,ILIST
      COMMON/NSERCH/DMIN,IEXIT,VDIFF,TOT,GS,GS0
      COMMON/IOS/ IOT,INPET,INP,IFR,ITO,IED,IOPT,ISCR
C+
      COMMON/QUAMOD/IQUAD,IFTP,IAPPR,MAJOR
C+
C
C        KODE=1: FIRST CALL     KODE>1: S;UBSEQUENT CALLS
C
      IF(KODE.NE.1) GOTO 70
C
C        INITIALIZATION FOR THE FIRST CALL ONLY
C
      ILIST=0
      KLIST=0
      LIMIT=0
      MAT=0
      NITER=0
      NLIST=0
      V01=0.0
      V02=0.0
      V03=0.0
      DMIN=1.0
      DO 20 I=1,N
      DO 10 J=1,N
 10   H(J,I)=0.0
      H(I,I)=1.0
      IGR(I)=0
 20   X1T(I)=0.0
      IF(IPRIN.EQ.1) GOTO 60
C
C        PRINT HEADER
C
      IF(ISENS.EQ.1) GOTO 40
      WRITE(IOT,30)
 30   FORMAT(//46H OPTIMIZATION BEGINS WITH FOLLOWING VARIABLES ,
     +13HAND GRADIENTS//6X,9HVARIABLES,17X,9HGRADIENTS)
      GOTO 60
 40   WRITE(IOT,50)
 50   FORMAT(//47H SENSITIVITY ANALYSIS WITH FOLLOWING VARIABLES ,
     +13HAND GRADIENTS//6X,9HVARIABLES,17X,9HGRADIENTS)
 60   KODE=2
      LDSERC=1
```

```
      RETURN
C
C        SET SOME CONSTANTS
C
 70   ZERO1=0.01
      IBNDRY=0
      ALP=0.5
      ZERO=0.000001
      MUNG=0
      KPRINT=0
      IEXIT=0
      NP=N+1
      N1=N-1
      NN=N*NP/2
C
C        LDSERC=1: FIRST FUNCTION EVALUATION
C              =2: OPTIMIZATION IN PROGRESS
C
      IF(LDSERC.EQ.2) GOTO 280
      V0=V
      V1T=V                                                          11-08-85
      DO 80 I=1,N
      X0(I)=U(I)
      X1T(I)=U(I)                                                    11-07-85
      G1T(I)=G(I)
 80   G0(I)=G(I)
C
C        PRINT THE STARTING POINT
C
      DO 90 I=1,N
      ALT=EXP(X0(I))
      G0II=G0(I)/ALT
90    WRITE(IOT,105)I,ALT,I,G0(I)
C00   FORMAT(4H   (,I2,2H):,G12.5,10X,1H(,I2,2H):,G12.5)
105   FORMAT(4H   (,I2,2H):,G12.5,10X,1H(,I2,2H):,G12.5,2X)
110   FORMAT(2X,8HERR. F.=F14.7)
120   FORMAT(9X,12H----****----)
      WRITE(IOT,110)V0
C
C        IPRIN=1: PRINTOUTS AT EXPERT LEVEL
C             =0: PRINTOUTS AT NON-EXPERT LEVEL
C
      IF(IPRIN.NE.1) WRITE(IOT,120)
C
C     ISENS=1 INDICATES SENSITIVITY ANALYSIS, ONLY EXIT
C
      IF(ISENS.NE.1)GOTO 130
      ISENS=2
      RETURN
C
130   IF(IPRIN.NE.1) WRITE(IOT,140)
      IF(IPRIN.EQ.1) WRITE(IOT,150)
140   FORMAT(38H HOW MANY ITERATIONS BEFORE NEXT STOP
     +32H('0' RESULTS IN FINAL ANALYSIS.) ,/20H WANT INTERMEDIATE P,
     +41HRINTS (YES=1,NO=0). TYPE TWO NUMBERS; I,J    )
```

```
150   FORMAT(47H NO. OF ITERATIONS? PRINT INTERMEDIATE RESULTS?  ,
     + 10H(Y=1,N=0) )
      CALL TAKEIN(INPET,NOITEM)
      LIMIT=ARRAY(1)
      NLIST=ARRAY(2)
      ILIST=1-NLIST
      NITER=0
      IF(LIMIT.GT.0) GOTO 160
      IEXIT=1
      IFLAG=1
      GOTO 510
C
C        ITERATIVE LOOP STARTS HERE
C
160   NITER=NITER+1
C
C        SOLVE FOR GA= - H*GO, H STORES THE LDL FACTORS
C
C        FORWARD SUBSTITUTION, XA IS USED FOR TEMPORARY STORAGE
C
      XA(1)=-GO(1)
      IF(N.EQ.1) GOTO 190
      DO 180 I=2,N
      I1=I-1
      Z=-GO(I)
      DO 170 J=1,I1
170   Z=Z-H(I,J)*XA(J)
180   XA(I)=Z
C
C        BACKWARD SUBSTITUTION
C
190   GA(N)=XA(N)/H(N,N)
      IF(N.EQ.1) GOTO 220
      N1=N-1
      DO 210 I=1,N1
      I1=N-I
      I2=I1+1
      Z=0.0
      DO 200 J=I2,N
200   Z=Z+H(I1,J)*GA(J)
210   GA(I1)=XA(I1)/H(I1,I1)-Z
C
C        CHECK THE DIRECTIONAL GRADIENT
C        EXIT IF IT IS POSITIVE OR SUFFICIENTLY CLOSE TO ZERO
C
220   GS=0.0
      DO 230 I=1,N
230   GS=GS+GA(I)*GO(I)
      IF(GS.LT.-ZERO) GOTO 240
      IEXIT=2
      GOTO 460
C
240   GSO=GS
      ALAMDA=-2.0*V/GS
      IF(ALAMDA.GT.1.0) ALAMDA=1.0
```

```
      TOT=0.0
      VDIFF=V0
C+
      MAJOR=1
C+
250   IEXIT=3
C
C        COMPUTE THE INCREMENTS DX.  EXIT IF THEY ARE ALL VERY SMALL
C        OTHERWISE A NEW TRIAL POINT IS GIVEN BY X1T=X0+DX
C
      DO 260 I=1,N
      DX=ALAMDA*GA(I)
      IF(ABS(DX).GT.ZERO) IEXIT=0
      IF(IGR(I).EQ.0) GOTO 260                                          12-18-85
      TEMP=IGR(I)*DX                                                    12-18-85
      IF(TEMP.GT.0) DX=0.0                                              12-18-85
      IF(TEMP.LT.0) IGR(I)=0                                            12-18-85
260   X1T(I)=X0(I)+DX
      IF(IEXIT.EQ.3) GOTO 460
C
C        RETURN TO EVALUATE FUNCTION VALUE AND GRADIENTS AT X1T
C
      DO 270 I=1,N
C70   IF(G(I).NE.1.0E-6) U(I)=X1T(I)
270   U(I)=X1T(I)                                                       12-18-85
      MUNG=2
      LDSERC=2
      RETURN
C
280   V1T=V
C+
      MAJOR=0
C+
      IBNDRY=1
      DO 290 I=1,N
CBOB  IF(G(I).NE.1.0E-6) IBNDRY=0
      IF(IGR(I).EQ.0) IBNDRY=0                                           2-18-86
290   G1T(I)=G(I)
C
C        IBNDRY=1: ALLVARIABLES ARE FROZEN AT BOUNDARY, EXIT
C
      IF(IBNDRY.EQ.0) GOTO 300
      IEXIT=4
      GOTO 570
300   IF(V1T.GT.CONV) GOTO 310
C
C        THE FUNCTION VALUE FALLS BELOW CONVERGENCE THRESHOLD, EXIT
C
      IEXIT=5
      GOTO 460
C
C        COMPUTE THE DIRECTIONAL GRADIENT AT THE TRIAL POINT X1T
C
310   GYS=0.0
      DO 320 I=1,N
```

```
320   GYS=GYS+GA(I)*G1T(I)
C
C
C        TERMINATE THE LINE SEARCH IF THE DIRECTIONAL GRADIENT
C           HAS DECREASED BY MORE THAN TEN PERCENT
C        OTHERWISE ESTIMATE A NEW TRIAL POINT BY LINEAR EXTRAPOLATION
C           OR CUBIC INTERPOLATION AS APPROPRIATE
C
      IF(V1T.GE.V0) GOTO 350
CBOB  IF(ABS(GYS/GS0).LE.0.9) GOTO 360
      IF(ABS(GYS).LE.0.9*ABS(GS0))  GOTO 360                          12-18-85
      IF(GYS.GT.0.0) GOTO 350
C
C        LINEAR EXTRAPOLATION
C
      TOT=TOT+ALAMDA
      Z=10.0
      IF(GS.GE.GYS) GOTO 330
      Z1=10.0*ABS(GS-GYS)
      Z2=ABS(GYS)
      IF(Z2.GT.Z1) GOTO 330
      Z=10.0*Z2/Z1
330   ALAMDA=ALAMDA*Z
      V0=V1T
      GS=GYS
      DO 340 I=1,N
340   X0(I)=X1T(I)
      GOTO 250
C
C     CUBIC INTERPOLATION
C
350   Z=3.0*(V0-V1T)/ALAMDA+GYS+GS
      ZZ=SQRT(Z*Z-GS*GYS)
      GZ=GYS+ZZ
      Z=1.0-(GZ-Z)/(ZZ+GZ-GS)
      ALAMDA=ALAMDA*Z
      GOTO 250
C
C     LINE SEARCH COMPLETED.  THE MATRIX H SHOULD BE UPDATED.
C     NOTE:  ARRAYS XA, GA, AND X1 ARE USED AS WORKING SPACE.
C
360   ALAMDA=ALAMDA+TOT
      V0=V1T
      VDIFF=VDIFF-V0
      DGS=GYS-GS0
      LINK=1
C
C     CHOOSE FROM THE DFP AND THE COMPLEMENTARY DFP FORMULAS.
C     UPDATING H, BASED ON THE FOLLOWING TEST:
C
      IF(DGS+ALAMDA*GS0.GT.0.0) GOTO 380
C
      DO 370 I=1,N
370   GA(I)=G1T(I)-G0(I)
      SIG=1.0/(ALAMDA*DGS)
```

```
      GOTO 400
C
380   ZZ=ALAMDA/(DGS-ALAMDA*GS0)
      Z=DGS*ZZ-1.0
      DO 390 I=1,N
390   GA(I)=Z*G0(I)+G1T(I)
      SIG=1.0/(ZZ*DGS*DGS)
C
C     FORWARD SUBSTITUTION FOR XA=L*GA, WHERE H=LDL AND GA IS
C     COMPUTED ABOVE.
C
400   XA(1)=GA(1)
      DO 420 I=2,N
      I1=I-1
      Z=GA(I)
      DO 410 J=1,I1
410   Z=Z-H(I,J)*XA(J)
420   XA(I)=Z
C
C     UPDATE DIAGONAL ELEMENTS.  POSITIVE DEFINITENESS IN INSURED.
C
      DO 430 I=1,N
      Z=H(I,I)+SIG*XA(I)*XA(I)
      IF(Z.LE.0.0) Z=DMIN
      IF(Z.LT.DMIN) DMIN=Z
      H(I,I)=Z
      X1(I)=XA(I)*SIG/Z
430   SIG=SIG-X1(I)*X1(I)*Z
C
C     UPDATE THE L MATRIX
C
      N1=N-1
      DO 440 I=1,N1
      I1=I+1
      DO 440 J=I1,N
      GA(J)=GA(J)-H(I,J)*XA(I)
      H(I,J)=H(I,J)+X1(I)*GA(J)
440   H(J,I)=H(I,J)
C
      IF(LINK.EQ.2) GOTO 460
      LINK=2
      DO 450 I=1,N
450   GA(I)=G0(I)
      SIG=1.0/GS0
      IF(DGS+ALAMDA*GS0.GT.0.0) SIG=-ZZ
      GOTO 400
C
C     UPDATING OF H COMPLETED.
C
460   V0=V1T
      V03=V02
      V02=V01
      V01=V0
      DO 470 I=1,N
      X0(I)=X1T(I)
```

```
470   GO(I)=G1T(I)
      IF(NITER.GE.LIMIT.OR.IEXIT.GT.0) ILIST=0
      IF(ILIST.NE.0) GOTO 490
C
C     ILIST=0: PRINT VARIABLES AND GRADIENTS
C
      DO 480 I=1,N
      ALT=EXP(XO(I))
      GOII=GO(I)/ALT
CBOB  IF(GO(I).EQ.1.0E-6) ALT=EXP(U(I))
      IF(IGR(I).NE.0) ALT=EXP(U(I))
480   WRITE(IOT,105)I,ALT,I,GO(I)
490   WRITE(IOT,110) VO
      IF(IPRIN.NE.1.AND.ILIST.EQ.0) WRITE(IOT,120)
C
      INEXP=1
C     IF INEXP=0, CHECK FOR STRICT DECREASE IN FUNCTION VALUE
C     AND ENABLE FRACTIONAL TERMINATION
C
      IF(INEXP.EQ.1) GOTO 500
      VOO=ABS(VO2-VO1)
      V1P=VO2*0.01
      IF(VOO.LT.V1P) IEXIT=6
      VOO=ABS(VO3-VO1)
      V4P=VO3*0.04
      IF(VOO.LT.V4P) IEXIT=6
C
500   IF(IEXIT.GT.0) GOTO 510
      IF(NITER-LIMIT) 160,130,130
C
C     EXIT.  SET LDSERC=-1, KODE=4, AND MUNG=1.  THE MAIN PROGRAM
C     WILL DETECT LDSERC<0.
C
510   IF(IEXIT.EQ.1) WRITE(IOT,520)
      IF(IEXIT.EQ.2) WRITE(IOT,530)
      IF(IEXIT.EQ.3) WRITE(IOT,540)
      IF(IEXIT.EQ.5) WRITE(IOT,550)
      IF(IEXIT.EQ.6) WRITE(IOT,560)
520   FORMAT(43H SEARCH INTERRUPTED, FINAL ANALYSIS FOLLOWS  /)
530   FORMAT(47H GRADIENT TERMINATION WITH ABOVE VALUES. FINAL  ,
     *16HANALYSIS FOLLOWS /)
540   FORMAT(44H LOCAL MIN REACHED.  GRADIENT SEARCH CANNOT   ,
     *7HIMPROVE)
550   FORMAT(47H FUNCTION TERMINATION WITH ABOVE VALUES. FINAL ,
     *16HANALYSIS FOLLOWS /)
560   FORMAT(49H FRACTIONAL TERMINATION WITH ABOVE VALUES. FINAL  ,
     *16HANALYSIS FOLLOWS /)
      GOTO 610
C
570   DO 580 I=1,N
      ALT=EXP(U(I))
580   WRITE(IOT,590) I, ALT
590   FORMAT(4H   (,I2,2H):,G12.5)
      WRITE(IOT,110) V
      IF(IPRIN.NE.1) WRITE(IOT,120)
```

```
      WRITE(IOT,600)
600   FORMAT(51H ALL VARIABLES AT BOUNDARY, FINAL ANALYSIS FOLLOWS /)
C
610   KODE=4
      MUNG=1
      LDSERC=-1
      RETURN
      END
```

```
      SUBROUTINE FTP
C
      DIMENSION X(15),CX(15),V(210),AF(1),TAF(1),H(15)
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,W,FR,NT,
     +NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON/CO2/SEND(15),GR(15),ERR,NOPT,CONV,INEXP,IGR(15)
      COMMON/CO3/CARAY(30,2,2),DATX(70,6)
      COMMON/ANALZ/IBAND(3),RIND(70),INC1(70),INC2(70),INC3(70),KX(15),
     +KY(15)
      COMMON/TOLER/NVERT,TOL(15),NRANK,NCOUNT,NCALL,TRES,WIT(65)
      COMMON/QUAMOD/IQUAD,IFTP,IAPPR,MAJOR
      COMMON/MIM/B(136,1),IV(6,64)
      COMPLEX CARAY
C
C     THIS SUBROUTINE IS DESIGNED TO SOLVE SEVERAL VERSIONS OF THE
C     FIXED TOLERANCE PROBLEM (FTP). CURRENTLY AVAILABLE ARE VERSIONS
C     A (FTP WITH ALL VERTICES OF THE TOLERANCE REGION), B (FTP
C     WITH A SPECIFIED NUMBER OF WORST VERTICES, C (FTP WITH A
C     THRESHOLD VALUE OF THE OBJECTIVE FUNCTION AND D (FTP WITH
C     WEIGHTED VERTICES PLUS WEIGHTED NOMINAL). SOME OF THE VERSIONS
C     CAN USE QUADRATIC APPROXIMATION WHICH IS DEVELOPED WHEN THE
C     SUBROUTINE FTP IS CALLED FOR THE FIRST TIME (NCALL=1).
C
      DATA V/-0.9,-0.7,-0.5,-0.3,-0.1,0.1,0.3,0.5,0.7,0.9,
     +     -0.8,-0.6,-0.4,-0.2,-0.1,0.1,0.2,0.4,0.6,0.8,
     +     -0.85,-0.65,-0.45,-0.25,-0.05,0.05,0.25,0.45,0.65,0.85,
     +      180*0.0/
      IF(KODE.NE.1)GO TO 15
      PRINT *,"DO YOU WANT TO USE QUADRATIC APPROXIMATION FOR THE FIXED
     +TOLERANCE PROBLEM (1/0)",
      READ(5,*)IQUAD
      IF(IQUAD.EQ.1) PRINT(6,1100)
1100  FORMAT(1X," THE QUADRATIC APPROXIMATION DEVELOPED IN THIS RUN"
     +/" MAY BE USED FOR ALL VERSIONS OF THE FTP, HOWEVER AT THIS TIME"
     +/" THE USE OF QUADRATIC MODEL IS IMPLEMENTED ONLY FOR VERSION 1."
     +/)
      PRINT *,"ENTER TOLERANCES FOR ALL COMPONENTS TOL(1)...TOL(NOPT)",
      READ(5,*) (TOL(I),I=1,NOPT)
      DO 7 I=1,NOPT
      H(I)=TOL(I)*DATX(KX(I),KY(I))
7     CONTINUE
      IF(IQUAD.NE.1)GO TO 99
C
C     THE QUADRATIC APPROXIMATION OPTION HAS BEEN SELECTED HERE AND
C     THE FOLLOWING LINES PREPARE DATA FOR CALCULATING QUADRATIC MODELS.
C
      N=NOPT
      NR=1
      NOBP=(NOPT+1)*(NOPT+2)/2
      MP=1
      NREM=NOPT*(NOPT-1)
      IJ=1
      DO 5 I=1,NOPT
      X(I)=DATX(KX(I),KY(I))
5     CONTINUE
```

```
      KODE=2
      CALL MODEL4(N,NR,NOBP,X,B,V,CX,AF,TAF,H,MP,NREM,IJ)
      KODE=1
      DO 9 I=1,NOPT
      DATX(KX(I),KY(I))=CX(I)
9     CONTINUE
C     WRITE(6,1000)(B(I,1),I=1,NOBP)
1000  FORMAT(5E16.8)
99    PRINT *,"SELECT VERSION OF THE FTP:
     +              1 - FTP WITH ALL VERTICES OF THE TOLERANCE REGION
     +                  2 - FTP WITH SPECIFIED NUMBER OF WORST VERTICES
     +                      3 - FTP WITH A THRESHOLD VALUE OF THE OBJECT
     +IVE FUNCTION               4 - FTP WITH WEIGHTED VERTICES PLUS WEI
     +GHTED NOMINAL",
      READ(5,*)IFTP
15    IF(IFTP.EQ.1)GO TO 10
      IF(IFTP.EQ.2)GO TO 20
      IF(IFTP.EQ.3)GO TO 30
      IF(IFTP.EQ.4)GO TO 35
      GO TO 40
10    CALL FTPAQ
      GO TO 40
20    CALL FTPBQ
      GO TO 40
30    CALL FTPCQ
      GO TO 40
35    IF(NOPT.EQ.6)GO TO 36
      PRINT(6,1200)NOPT
1200  FORMAT(1X," YOU HAVE SELECTED NOPT=",I3,1X,"VARIABLES."
     +/" THE NUMBER OF VARIABLES FOR VERSION 4 OF THE FTP MUST BE"
     +/" 6 AT THIS TIME. PLEASE SELECT ANOTHER OPTION.")
      GO TO 99
36    CALL FTPDQ
40    IF(MAJOR.EQ.1)NCALL=NCALL+1
      RETURN
      END
```

```
      SUBROUTINE FTPAQ
C
      DIMENSION ER(64),GE(12,64),H(15,15),Y(15)
      DIMENSION ZT(64,6)
      DIMENSION XNOM(15),XT(15),XX(15),DXDXO(15),X(30),GFTP(15)
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,W,FR,NT,
     +NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON /CO2/SEND(15),GR(15),ERR,NOPT,CONV,INEXP,IGR(15)
      COMMON /CO3/CARAY(30,2,2),DATX(70,6)
      COMMON /ANALZ/IBAND(3),RIND(70),INC1(70),INC2(70),INC3(70),KX(15),
     +KY(15)
      COMMON /TOLER/NVERT,TOL(15),NRANK,NCOUNT,NCALL,TRES,WIT(65)
      COMMON /QUAMOD/IQUAD,IFTP,IAPPR,MAJOR
      COMMON /MIM/B(136,1),IV(6,64)
      COMPLEX CARAY
      DATA ZP/1H+/,ZM/1H-/
C
      IF(KODE.NE.1)GO TO 15
      NVERT=2**NOPT
      IF(IQUAD.NE.1)GO TO 22
      PRINT *,"DO YOU WANT TO USE QUADRATIC APPROXIMATION FOR VERSION
     +A OF THE FTP(1/0)",
      READ(5,*)IAPPR
C
      IF(IAPPR.EQ.1)PRINT(6,1000)
1000  FORMAT(1X," THE OPTION YOU HAVE SELECTED REQUIRES SOME"
     +/" COMMENTS. SINCE THE QUADRATIC MODEL DEVELOPED EARLIER IN"
     +/" THIS RUN IS VALID ONLY FOR A FEW ITERATIONS IT IS RECOMMENDED"
     +/" THAT YOU RUN OPTIMIZATION ONLY FOR 5 ITERATIONS. IN ORDER"
     +/" TO CONTINUE OPTIMIZATION USING QUADRATIC APPROXIMATION QUIT"
     +/" AFTER 5 ITERATIONS, SAVE RESULTS AND RESTART THE PROCESS"
     +/" AGAIN")
22    DO 20 I=1,NOPT
      XNOM(I)=DATX(KX(I),KY(I))
20    CONTINUE
      GO TO 25
15    DO 35 I=1,NOPT
      XNOM(I)=EXP(SEND(I))
35    CONTINUE
25    DO 10 I=1,NOPT
      XT(I)=TOL(I)*XNOM(I)
      GFTP(I)=0.0
10    CONTINUE
      EFTP=0.0
      IF(IAPPR.EQ.1)GO TO 999
      CALL VERT(IV,NOPT,NVERT)
      DO 40 J=1,NVERT
      DO 30 K=1,NOPT
      Z=ZP
      IF(IV(K,J).EQ.-1)Z=ZM
      DXDXO(K)=1.0+IV(K,J)*TOL(K)
      XX(K)=XNOM(K)*DXDXO(K)
      ZT(J,K)=Z
30    CONTINUE
      IF(KODE.EQ.1)GO TO 13
```

```
      DO 70 I=1,NOPT
      SEND(I)=ALOG(XX(I))
70    CONTINUE
      GO TO 100
13    DO 45 I=1,NOPT
      DATX(KX(I),KY(I))=XX(I)
45    CONTINUE
C
100   CALL ANALYZ
C
      EFTP=EFTP+ERR
      DO 50 JJ=1,NOPT
      GFTP(JJ)=GFTP(JJ)+GR(JJ)
50    CONTINUE
40    CONTINUE
      GO TO 555
C
C     HERE WE USE QUADRATIC APPROXIMATION AND THE FOLLOWING PART
C     OF THE PROGRAM PREPARES DATA FOR QUADRATIC POLYNOMIAL EVALUATION
C
999   N=NOPT+NOPT
      NN=NOPT
      NAC=1
      DO 110 I=1,NOPT
      X(I)=XNOM(I)
      X(I+NN)=XT(I)
110   CONTINUE
      NOBP=(NOPT+1)*(NOPT+2)/2
      NV=NVERT
      LL=1
      CALL VERT(IV,NN,NV)
      CALL QPE(N,NN,NAC,X,ER,GE,B,IV,H,Y,NOBP,NV,LL)
C     WRITE(6,*)(ER(I),I=1,NVERT)
      DO 120 I=1,NVERT
      EFTP=EFTP+ER(I)
      DO 130 J=1,NOPT
      GFTP(J)=GFTP(J)+GE(J,I)*(1.0+IV(J,I)*TOL(J))*XNOM(J)
130   CONTINUE
120   CONTINUE
555   DO 55 I=1,NOPT
      SEND(I)=ALOG(XNOM(I))
55    CONTINUE
      ERR=EFTP
      DO 60 I=1,NOPT
      GR(I)=GFTP(I)
60    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FTPBQ
C
      DIMENSION XT(15),XNOM(15),XX(15),GFTP(15),FERR(64),NRF(64)
      DIMENSION ZT(64,6),DXDXO(15)
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,
     *W,FR,NT,NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON/CO2/SEND(15),GR(15),ERR,NOPT,CONV,INEXP,IGR(15)
      COMMON /CO3/ CARAY(30,2,2),DATX(70,6)
      COMMON/ANALZ/IBAND(3),RIND(70),INC1(70),INC2(70),INC3(70),
     *KX(15),KY(15)
      COMMON/TOLER/NVERT,TOL(15),NRANK,NCOUNT,NCALL,TRES,WIT(65)
      COMMON/MIM/B(136,1),IV(6,64)
      COMPLEX CARAY
      DATA ZP/1H+/,ZM/1H-/
C
C     THE FIXED TOLERANCE PROBLEM IS ORGANIZED AROUND THE CURRENT VALUE
C     OF THE NOMINAL POINT XNOM(I), I=1,...,NOPT,AND ONLY A LIMITED
C     NUMBER OF VERTICES (NRANK) OF THE TOLERANCE REGION (THE NUMBER
C     OF ALL VERTICES IS NVERT) IS CONSIDERED
C     WRITE(6,*)KODE,NRANK
C
666   NCOUNT=NCOUNT+1
      IF(KODE.NE.1) GOTO 15
      IF(IQUAD.EQ.1)PRINT(6,1600)
1600  FORMAT(1X," YOU HAVE SELECTED VERSION 2 OF THE FTP. NOTE,"
     +/" THAT AT THIS TIME THE QUADRATIC APPROXIMATION DEVELOPED"
     +/" EARLIER IN THIS RUN WILL NOT BE USED FOR OPTIMIZATION."
     +/" ALL FUNCTION VALUE CALCULATIONS ARE EXACT."/)
      NVERT=2**NOPT
      PRINT(6,1700)
1700  FORMAT(1X," WE RECOMMEND THAT THE NUMBER OF WORST VERTICES "
     +/" YOU ARE ABOUT TO ENTER SHOULD BE AT LEAST 2/3 OF THE NUMBER"
     +/" OF ALL POSSIBLE VERTICES OF THE TOLERANCE REGION TO OBTAIN"
     +/" REASONABLE RESULTS."/)
      PRINT*," ENTER THE NUMBER OF WORST VERTICES FOR THE FTP",
      READ(5,*)NRANK
C
C     WHEN THE SUBROUTINE FTPBQ IS CALLED FOR THE FIRST TIME, KODE=1 AND
C     OPTIMIZATION DATX(KX(I),KY(I)) READ FROM THE DATA FILE.
C
      DO 20 I=1,NOPT
C     WRITE(6,1500)I,DATX(KX(I),KY(I))
1500  FORMAT(3X," I=",I3,2X," DATX(I)=",E12.7)
      XNOM(I)=DATX(KX(I),KY(I))
20    CONTINUE
      GOTO 25
C
C     WHEN THE SUBROUTINE "FTPBQ" IS CALLED FOR THE SECOND AND SUBSEQUENT
C    TIMES, KODE IS NOT 1 AND THE NOMINAL POINT HAS TO BE RECOVERED
C     FROM SEND(I) THROUGH A REVERSE TRANSFORMATION.
C
15    DO 35 I=1,NOPT
      XNOM(I)=EXP(SEND(I))
35    CONTINUE
C
```

```
C     THE VECTOR OF ABSOLUTE TOLERANCES XT(I) IS EVALUATED BASED ON
C     THE VALUE OF THE RELATIVE TOLERANCE TOL.  IN THIS VERSION OF THE
C     "FTPBQ" SUBROUTINE THE SAME RELATIVE TOLERANCE IS ASSUMED FOR ALL
C     CIRCUIT ELEMENTS.  ALSO, THE OBJECTIVE FUNCTION VALUE "EFTP" AND
C     ITS GRADIENT W.R.T. ALL VARIABLES ARE INITIALIZED.
C
25    DO 10 I=1,NOPT
      XT(I)=TOL(I)*XNOM(I)
      GFTP(I)=0.0
10    CONTINUE
      EFTP=0.0
C
C     DO LOOP 40 GENERATES ALL VERTICES OF THE TOLERANCE REGION AND
C     FOR EACH OF THE GENERATED VERTICES THE SUBROUTINE "ANALYZ" IS
C     CALLED WHICH CALCULATES THE OBJECTIVE FUNCTION COMPONENT (AND
C     ITS GRADIENT) CORRESPONDING TO ONE VERTEX AT ALL FREQUENCIES
C     DEFINED FOR THE PROBLEM.
C     WHEN KODE IS NOT 1 ONLY NRANK VERTICES ARE CONSIDERED AND THOSE
C     VERTICES ARE THE FIRST NRANK VERTICES STORED IN THE VECTOR NRF(I)
C     THAT IS CREATED DURING THE SECOND CALL TO FTPBQ.
C
C     WRITE(6,*)KODE,NVERT,NRANK
      CALL VERT(IV,NOPT,NVERT)
      DO 40 J=1,NVERT
      IF(NCOUNT.LT.3)GO TO 46
      DO 44 I=1,NRANK
      IF(J.EQ.NRF(I))GO TO 46
44    CONTINUE
      GO TO 40
46    DO 30 K=1,NOPT
      Z=ZP
      IF(IV(K,J).EQ.-1)Z=ZM
      DXDXO(K)=1.0+IV(K,J)*TOL(K)
      XX(K)=XNOM(K)*DXDXO(K)
      ZT(J,K)=Z
30    CONTINUE
C     WRITE(6,1900)(XX(I),I=1,NOPT)
1900  FORMAT(7X," XX(I)=",6(E12.7,2X))
C     WRITE(6,*)KODE
      IF(KODE.EQ.1) GOTO 13
C
C     IF KODE IS NOT 1 (WHICH IS THE CASE AFTER "SEARCH" HAS BEEN CALLED
C     AT LEAST ONCE) THEN THE GENERATED VERTEX XX(I) IS TRANSFORMED
C     AND ASSIGNED TO SEND(I) WHICH IN TURN IS PROCESSED IN "ANALYZ"
C     TO FORM A NEW "DATX(KX(I),KY(I))" NECESSARY FOR RESPONSE CALCULATIONS.
C
      DO 70 I=1,NOPT
      SEND(I)=ALOG(XX(I))
70    CONTINUE
      GOTO 100
C
C     KODE=1 (WHICH IS THE CASE BEFORE THE SUBROUTINE "SEARCH" IS
C     CALLED FOR THE FIRST TIME) THEN THE GENERATED VERTEX XX(I)
C     IS ASSIGNED TO THE VECTOR "DATX(KX(I),KY(I))" USED IN "ANALYZ" TO
C     CALCULATE THE RESPONSES.
```

```
C
13    DO 45 I=1,NOPT
      DATX(KX(I),KY(I))=XX(I)
45    CONTINUE
C
100   CALL ANALYZ
C     WRITE(6,1200)J,ERR
1200  FORMAT(2X," J=",I3,2X," ERR=",F12.7)
      FERR(J)=ERR
      EFTP=EFTP+ERR
      DO 50 JJ=1,NOPT
      GFTP(JJ)=GFTP(JJ)+GR(JJ)
50    CONTINUE
40    CONTINUE
C     DURING THE SECOND CALL TO FTPB THE SUBROUTINE ANALYZ IS CALLED
C     FOR ALL VERTICES OF THE TOLERANCE REGION AND ERROR FUNCTION VALUES
C     CORRESPONDING TO ALL VERTICES ARE STORED IN THE VECTOR FERR(I).
C     THEN ALL COMPONENTS OF THE VECTOR FERR(I) ARE ORDERED IN DESCENDING
C     ORDER (ACCORDING TO THEIR VALUE) AND VECTOR NRF(I) CONTAINS THE
C     ORIGINAL INDICES (VERTEX NUMBERS) OF THE ORDERED ERROR FUNCTIONS.
C
      IF(NCOUNT.GT.2)GO TO 56
      DO 33 I=1,NVERT
      NRF(I)=I
33    CONTINUE
      NVERT1=NVERT-1
      DO 61 J=1,NVERT1
      FMAX=0.0
      JMAX=0
      DO 51 I=J,NVERT
      IF(FERR(I).LE.FMAX)GO TO 51
      FMAX=FERR(I)
      JMAX=NRF(I)
      K=I
51    CONTINUE
      NRF(K)=NRF(J)
      NRF(J)=JMAX
      FERR(K)=FERR(J)
      FERR(J)=FMAX
61    CONTINUE
      DO 111 I=1,NVERT
      IPR=NRF(I)
      IF(KODE.EQ.2)WRITE(6,999)I,IPR,FERR(I),(ZT(NRF(I),J),J=1,NOPT)
999   FORMAT(2X," I=",I3,2X," VERT.NO.=",I3,2X," ERR=",F12.7,2X,6A1)
111   CONTINUE
C
C     AFTER ALL VERTICES HAVE BEEN EXHAUSTED (DO LOOP 40 COMPLETED)
C     THE VECTOR OF OPTIMIZATION VARIABLES "SEND(I)" (CONTAINING THE
C     PARAMETER VALUES CORRESPONDING TO THE LAST VERTEX UPON THE
C     COMPLETION OF LOOP 40) HAS TO BE RESTORED TO CONTAIN THE NOMINAL
C     POINT "XNOM(I)" WHICH IS THE ACTUAL OPTIMIZATION VARIABLE OF THE
C     PROBLEM.
C
56    DO 55 I=1,NOPT
      SEND(I)=ALOG(XNOM(I))
```

```
55    CONTINUE
      IF(NCOUNT.EQ.2)GO TO 666
C
C     THE VARIABLE "ERR" AND THE VECTOR "GR(I)" ARE ASSIGNED THE VALUES
C     OF "EFTP" AND "GFTP(I)" WHICH ARE THE OVERALL OBJECTIVE FUNCTION
C     VALUE AND ITS GRADIENT FOR THE FIXED TOLERANCE PROBLEM, RESPECTI-
C     VELY.
C
      ERR=EFTP
      DO 60 I=1,NOPT
      GR(I)=GFTP(I)
60    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FTPCQ
C
      DIMENSION ER(64),GE(12,64),H(15,15),Y(15),FERR(64),NRF(64)
      DIMENSION ZT(64,6)
      DIMENSION XNOM(15),XT(15),XX(15),DXDXO(15),X(30),GFTP(15)
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,W,FR,NT,
     +NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON /CO2/SEND(15),GR(15),ERR,NOPT,CONV,INEXP,IGR(15)
      COMMON /CO3/CARAY(30,2,2),DATX(70,6)
      COMMON /ANALZ/IBAND(3),RIND(70),INC1(70),INC2(70),INC3(70),KX(15),
     +KY(15)
      COMMON /TOLER/NVERT,TOL(15),NRANK,NCOUNT,NCALL,TRES,WIT(65)
      COMMON /QUAMOD/IQUAD,IFTP,IAPPR,MAJOR
      COMMON /MIM/B(136,1),IV(6,64)
      COMMON/LAST/XNOMLST(15),NRFLST(64),FERRLST(64),ZTLST(64,6)
      COMPLEX CARAY
      DATA ZP/1H+/,ZM/1H-/
C
      IF(MAJOR.NE.1)GO TO 117
      DO 111 I=1,NVERT
      WRITE(6,888)I,NRFLST(I),FERRLST(I),(ZTLST(NRFLST(I),J),J=1,NOPT)
888   FORMAT(2X," I=",I3,2X," VERT.NO.=",I3,2X," ERR=",F12.7,2X,6A1)
111   CONTINUE
117   IF(KODE.NE.1)GO TO 15
      NVERT=2**NOPT
      IF(IQUAD.EQ.1)PRINT(6,1000)
1000  FORMAT(1X,/" YOU HAVE SELECTED VERSION 3 OF THE FTP. NOTE,"
     +/" THAT AT THIS TIME THE QUADRATIC MODEL DEVELOPED EARLIER IN"
     +/" THIS RUN WILL NOT BE  USED FOR OPTIMIZATION. ALL FUNCTION"
     +/" VALUE CALCULATIONS ARE EXACT."/)
      PRINT*," ENTER THRESHOLD VALUE FOR THE FUNCTION CHARACTERIZING A G
     +IVEN VERTEX. FROM OUR COMPUTATIONAL EXPERIENCE WE RECOMMEND THAT
     + THE THRESHOLD VALUE BE 5*OBJ. FUNCTION VALUE AT THE SOLUTION TO
     + THE NOMINAL DESIGN PROBLEM",
      READ(5,*)TRES
C     IF(IQUAD.NE.1)GO TO 22
C     PRINT *,"DO YOU WANT TO USE QUADRATIC APPROXIMATION FOR THE FTP
C    +(1/0)",
C     READ(5,*)IAPPR
C
      DO 20 I=1,NOPT
      XNOM(I)=DATX(KX(I),KY(I))
20    CONTINUE
      GO TO 25
15    DO 35 I=1,NOPT
      XNOM(I)=EXP(SEND(I))
35    CONTINUE
25    DO 10 I=1,NOPT
      XT(I)=TOL(I)*XNOM(I)
      GFTP(I)=0.0
10    CONTINUE
      EFTP=0.0
C     IF(IAPPR.EQ.1)GO TO 999
      CALL VERT(IV,NOPT,NVERT)
      DO 40 J=1,NVERT
```

```
      DO 30 K=1,NOPT
      Z=ZP
      IF(IV(K,J).EQ.-1.0)Z=ZM
      DXDXO(K)=1.0+IV(K,J)*TOL(K)
      XX(K)=XNOM(K)*DXDXO(K)
      ZT(J,K)=Z
30    CONTINUE
      IF(KODE.EQ.1)GO TO 13
      DO 70 I=1,NOPT
      SEND(I)=ALOG(XX(I))
70    CONTINUE
      GO TO 100
13    DO 45 I=1,NOPT
      DATX(KX(I),KY(I))=XX(I)
45    CONTINUE
C
100   CALL ANALYZ
      FERR(J)=ERR
      IF(ERR.GT.TRES)GO TO 53
      GO TO 40
53    EFTP=EFTP+(ERR-TRES)
C
      DO 50 JJ=1,NOPT
      GFTP(JJ)=GFTP(JJ)+GR(JJ)
50    CONTINUE
40    CONTINUE
      IF(KODE.NE.2)GO TO 666
      DO 33 I=1,NVERT
      NRF(I)=I
33    CONTINUE
      NVERT1=NVERT-1
      DO 61 J=1,NVERT1
      FMAX=0.0
      JMAX=0
      DO 51 I=J,NVERT
      IF(FERR(I).LE.FMAX)GO TO 51
      FMAX=FERR(I)
      JMAX=NRF(I)
      K=I
51    CONTINUE
      NRF(K)=NRF(J)
      NRF(J)=JMAX
      FERR(K)=FERR(J)
      FERR(J)=FMAX
61    CONTINUE
C
666   DO 55 I=1,NOPT
      SEND(I)=ALOG(XNOM(I))
55    CONTINUE
      ERR=EFTP
      DO 60 I=1,NOPT
      GR(I)=GFTP(I)
60    CONTINUE
      IF(KODE.NE.2)GO TO 699
      DO 150 J=1,NVERT
```

```
      DO 140 I=1,NOPT
      XNOMLST(I)=XNOM(I)
      ZTLST(J,I)=ZT(J,I)
140   CONTINUE
      NRFLST(J)=NRF(J)
      FERRLST(J)=FERR(J)
150   CONTINUE
699   RETURN
      END
```

```
      SUBROUTINE FTPDQ
C
      DIMENSION ER(64),GE(12,64),H(15,15),Y(15),ZT(64,6)
      DIMENSION XNOM(15),XT(15),XX(15),DXDXO(15),X(30),GFTP(15)
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,W,FR,NT,
     +NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON /CO2/SEND(15),GR(15),ERR,NOPT,CONV,INEXP,IGR(15)
      COMMON /CO3/CARAY(30,2,2),DATX(70,6)
      COMMON /ANALZ/IBAND(3),RIND(70),INC1(70),INC2(70),INC3(70),KX(15),
     +KY(15)
      COMMON /TOLER/NVERT,TOL(15),NRANK,NCOUNT,NCALL,TRES,WIT(65)
      COMMON /QUAMOD/IQUAD,IFTP,IAPPR,MAJOR
      COMMON /MIM/B(136,1),IV(6,64)
      COMPLEX CARAY
C
C     IF(NCALL.GT.1)GO TO 15
      IF(KODE.NE.1)GO TO 15
      NVERT=2**NOPT
      IF(IQUAD.EQ.1) PRINT(6,1000)
1000  FORMAT(1X," YOU HAVE SELECTED VERSION 4 OF THE FTP. NOTE,"
     +/" THAT AT THIS TIME THE QUADRATIC APPROXIMATION DEVELOPED"
     +/" EARLIER IN THIS RUN WILL NOT BE USED FOR OPTIMIZATION."
     +/" ALL FUNCTION VALUE CALCULATIONS ARE EXACT.")
      DO 5 I=1,NVERT
      WIT(I)=1.0
5     CONTINUE
      WIT(65)=64
C
      DO 20 I=1,NOPT
      XNOM(I)=DATX(KX(I),KY(I))
20    CONTINUE
      GO TO 25
15    DO 35 I=1,NOPT
      XNOM(I)=EXP(SEND(I))
35    CONTINUE
25    DO 10 I=1,NOPT
      XT(I)=TOL(I)*XNOM(I)
      GFTP(I)=0.0
10    CONTINUE
      EFTP=0.0
C     IF(IAPPR.EQ.1)GO TO 999
      CALL VERT(IV,NOPT,NVERT)
      DO 40 J=1,NVERT
      DO 30 K=1,NOPT
      Z=ZP
      IF(IV(K,J).EQ.-1)Z=ZM
      DXDXO(K)=1.0+IV(K,J)*TOL(K)
      XX(K)=XNOM(K)*DXDXO(K)
      ZT(J,K)=Z
30    CONTINUE
      IF(KODE.EQ.1)GO TO 13
      DO 70 I=1,NOPT
      SEND(I)=ALOG(XX(I))
70    CONTINUE
      GO TO 100
```

```
13    DO 45 I=1,NOPT
      DATX(KX(I),KY(I))=XX(I)
45    CONTINUE
C
100   CALL ANALYZ
C
      EFTP=EFTP+WIT(J)*ERR
      DO 50 JJ=1,NOPT
      GFTP(JJ)=GFTP(JJ)+WIT(J)*GR(JJ)
50    CONTINUE
40    CONTINUE
      GO TO 555
C
555   DO 55 I=1,NOPT
      SEND(I)=ALOG(XNOM(I))
55    CONTINUE
      IF(KODE.NE.1)GO TO 333
      DO 69 I=1,NOPT
      DATX(KX(I),KY(I))=XNOM(I)
69    CONTINUE
333   CALL ANALYZ
      ERRNOM=ERR
      ERR=EFTP+ERRNOM*WIT(65)
      DO 60 I=1,NOPT
      GR(I)=GFTP(I)+GR(I)*WIT(65)
60    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MODEL4(N,NR,NOBP,X,B,V,CX,AF,TAF,H,MP,NREM,IJ)
C
C     THIS SUBROUTINE CALCULATES COEFFICIENTS OF QUADRATIC MODELS
C     OF SURFACES IN OPTIMAL DESIGN
C
C     N - NUMBER OF VARIABLES
C
C     NR - NUMBER OF FUNCTIONS TO BE INTERPOLATED
C
C     NOBP - NUMBER OF BASE POINTS=(N+1)*(N+2)/2
C
C     X - ARRAY OF DIMENSION N IN WHICH CENTER OF INTERPOLATION
C         IS GIVEN
C
C     B - ARRAY OF DIMENSIONS NOPB BY NR IN WHICH COEFFICIENTS
C         OF THE INTERPOLATING POLYNOMIALS ARE RETURNED, E.G.,
C         B(5,2) IS THE 5TH COEFFICIENT OF THE 2ND POLYNOMIAL
C
C     NREM = N*(N-1)
C
C     V - ARRAY OF DIMENSIONS NREM IN WHICH THE USER SUPPLIES
C         NUMBERS SUCH THAT ABS(V) IS LESS THAN ONE AND NOT EQUAL
C         TO ZERO. THESE NUMBERS ARE USED TO CHANGE TWO PARAMETERS
C         AT A TIME
C
C     CX - WORKING ARRAY OF DIMENSION N TO STORE THE CENTER OF
C          INTERPOLATION
C
C     AF - ARRAY OF DIMENSION NR USED TO STORE ACTUAL FUNCTION
C          VALUES AT A CERTAIN BASE POINT
C
C     TAF - ARRAY OF DIMENSION NR USED TO STORE 2*AF
C
C     H - ARRAY OF DIMENSION N IN WHICH THE STEPS DEFINING THE
C         SIZE OF THE INTERPOLATION REGION ARE SUPPLIED BY THE USER
C
C     MP - 1 IF QUADRATIC MODEL IS TO BE PRINTED, OTHERWISE SET TO
C          ZERO
C
C     IJ - 1 FOR THE FIRST CALL TO MODEL4, OTHERWISE SET TO ZERO
C
C     SUBROUTINE ACFUNS(X,AF) IS CALLED TO SUPPLY VALUES OF AF(NR)
C     AT POINT X(N)
C
C
      REAL X(N),B(NOBP,NR),V(NREM),CX(N),AF(NR),TAF(NR),H(N)
C
      IF(IJ.EQ.0) GO TO 10
      MM=((N+2)*(N+1))/2
      MM1=MM-1
      MMN=MM-N-1
      LK=N-1
10    DO 20 J=1,N
      CX(J)=X(J)
20    CONTINUE
```

```
      CALL ACFUNS(X,AF)
      DO 25 J=1,NR
      B(MM,J)=AF(J)
      TAF(J)=AF(J)+AF(J)
25    CONTINUE
      DO 40 I=1,N
      X(I)=CX(I)+H(I)
      CALL ACFUNS(X,AF)
      DO 30 J=1,NR
      B(I,J)=AF(J)
30    CONTINUE
      X(I)=CX(I)-H(I)
      CALL ACFUNS(X,AF)
      TH=H(I)+H(I)
      HH=H(I)*H(I)
      THH=HH+HH
      DO 35 J=1,NR
      B(MMN+I,J)=(B(I,J)-AF(J))/TH
      B(I,J)=(B(I,J)+AF(J)-TAF(J))/THH
35    CONTINUE
      X(I)=CX(I)
40    CONTINUE
      IR=N
      IRR=1
      DO 50 I=1,LK
      I1=I+1
      MI=MMN+I
      DO 50 K=I1,N
      VHI=V(IRR)*H(I)
      VHK=V(IRR+1)*H(K)
      MK=MMN+K
      IR=IR+1
      IRR=IRR+2
      X(I)=CX(I)+VHI
      X(K)=CX(K)+VHK
      DV=VHI*VHK
      CALL ACFUNS(X,AF)
      DO 45 J=1,NR
      B(IR,J)=(AF(J)-B(MM,J)-VHI*(VHI*B(I,J)+B(MI,J))
     +  -VHK*(VHK*B(K,J)+B(MK,J)))/DV
45    CONTINUE
      X(I)=CX(I)
      X(K)=CX(K)
50    CONTINUE
      DO 65 J=1,NR
      DO 55 I=1,N
      MI=MMN+I
      A=B(I,J)*CX(I)
      B(MM,J)=B(MM,J)+CX(I)*(A-B(MI,J))
      B(MI,J)=B(MI,J)-A-A
55    CONTINUE
      IR=N
      DO 60 I=1,LK
      MI=MMN+I
      I1=I+1
```

```
      DO 60 K=I1,N
      MK=MMN+K
      IR=IR+1
      A=CX(I)*B(IR,J)
      B(MM,J)=B(MM,J)+A*CX(K)
      B(MI,J)=B(MI,J)-CX(K)*B(IR,J)
      B(MK,J)=B(MK,J)-A
60    CONTINUE
65    CONTINUE
C
      IF(MP.NE.1) RETURN
      WRITE(6,200)
      DO 70 I=1,N
      WRITE(6,250) H(I)
70    CONTINUE
      DO 150 MA=1,NR
      WRITE(6,270)MA
      WRITE(6,280) (B(I,MA),I=1,MM)
150   CONTINUE
200   FORMAT(//4X,"STEPS AROUND CENTER OF INTERPOLATION")
250   FORMAT(4X,E16.8)
270   FORMAT(/,4X,"ER(",I2,")",/,4X,6("-"))
C 280 FORMAT(4(3X,E16.8),/,4(3X,E16.8),/,2(3X,E16.8)/)
280   FORMAT(4(3X,E16.8))
      RETURN
      END
```

```
      SUBROUTINE QPE(N,NN,NAC,X,ER,GE,B,IV,H,Y,NOBP,NV,LL)
C
C      N - NUMBER OF VARIABLES INCLUDING TOLERANCES
C
C      NN - NUMBER OF NOMINAL VALUES OF PARAMETERS (OR NUMBER
C           OF VARIABLES IN CASE OF A SINGLE POINT)
C
C      NAC - NUMBER OF ACTUAL CONSTRAINTS, E.G., IF THERE ARE FOUR
C            VERTICES FOR TWO CONSTRAINTS NAC=2, 8 ERROR FUNCTIONS
C            RESULT
C
C      X - INPUT VECTOR OF DIMENSION N
C
C      ER - ARRAY OF DIMENSION NAC IF LL=0, OTHERWISE (LL=1)OF
C           DIMENSION NV*NAC
C
C      GE - ARRAY OF DIMENSION (N,M) STORING GRADIENTS OF ERROR
C           FUNCTIONS; M IS THE NUMBER OF ERROR FUNCTIONS
C
C      B - ARRAY OF POLYNOMIAL COEFFICIENTS, DIMENSIONED (NOPB,NAC),
C          WHERE NOBP=(NN+1)*(NN+2)/2
C
C     IV - ARRAYOF DIMENSIONS (NN,NV) DEFINING VERTICES; IT IS
C            SUPPLIED BY THE USER; IF EVALUATION AT A SINGLE POINT IS
C            REQUIRED (LL=0) THEN NV=1 AND IV IS NOT SUPPLIED
C
C      H - WORKING ARRAY OF DIMENSIONS (NN,NN)
C
C      Y - WORKING ARRAY OF DIMENSION NN
C
C      NOBP = (NN+1)*(NN+2)/2 USER SUPPLIED
C           NN
C      NV=2  NUMBER OF VERTICES (EQUALS 1 FOR A SINGLE POINT)
C
C      LL - INTEGER SUPPLIED BY THE USER, IF LL=1 EVALUATION AT VERTICES
C           IS REQUIRED, IF LL=0 EVALUATION AT A SINGLE POINT IS
C           REQUIRED
C
      REAL    X(1),ER(1),GE(N,1),B(NOBP,NAC),H(NN,NN),Y(NN)
      INTEGER IV(NN,NV)
C
      MM=NOBP
      MM1=MM-1
      LK=NN-1
      IF(LL.EQ.1)GO TO 10
      DO 5 I=1,NN
      Y(I)=X(I)
5     CONTINUE
      GO TO 25
C
C      STARTING AT THE VERTEX Y(I)=X(I)-X(I+NN)  X(I+NN) IS EPS(I)
C
10    DO 20 I=1,NN
      Y(I)=X(I)-X(I+NN)
20    CONTINUE
```

```
25    DO 100 K1=1,NAC
      K=1+NV*(K1-1)
      IF(LL.NE.1)K=K1
      S=B(MM,K1)
      DO 30 I=1,NN
      T=B(I,K1)*Y(I)
      S=S+T*Y(I)
      GE(I,K)=T+T
      H(I,I)=B(I,K1)+B(I,K1)
30    CONTINUE
      KM=NN
      DO 40 I=1,LK
      LH=I+1
      DO 40 J=LH,NN
      KM=KM+1
      T=B(KM,K1)*Y(I)
      S=S+T*Y(J)
      GE(I,K)=GE(I,K)+B(KM,K1)*Y(J)
      GE(J,K)=GE(J,K)+T
      H(I,J)=B(KM,K1)
      H(J,I)=H(I,J)
40    CONTINUE
      DO 50 I=1,NN
      KM=KM+1
      S=S+B(KM,K1)*Y(I)
      GE(I,K)=GE(I,K)+B(KM,K1)
50    CONTINUE
      ER(K)=S
      IF(LL.NE.1)GO TO 100
C
C      OTHER VERTICES
C
      JU=1
      L=K
      LG=1
      MI=K-1
      DO 90 I=1,NN
      M=MI
      INN=I+NN
      EP=X(INN)
      TE=EP+EP
      DO 70 IR=1,NN
      H(IR,I)=TE*H(IR,I)
70    CONTINUE
      EPH=EP*H(I,I)
      DO 80 J=1,JU
      M=M+1
      L=L+1
      LG=LG+1
      ER(L)=ER(M)+TE*GE(I,M)+EPH
      DO 80 IR=1,NN
      GE(IR,L)=GE(IR,M)+H(IR,I)
80    CONTINUE
      JU=JU+JU
90    CONTINUE
```

```
      L=K
      DO 99 II=1,NV
      DO 95 I=1,NN
      J=I+NN
      GE(J,L)=FLOAT(IV(I,II))*GE(I,L)
95    CONTINUE
      L=L+1
99    CONTINUE
100   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VERT(IV,NN,NV)
C
C     THIS SUBROUTINE GENERATES THE MU'S ACCORDING TO
C     THE COMMON NUMBERING SCHEME FOR THE VERTICES
C
C     NN - NUMBER OF PARAMETERS (INPUT)
C     NV - NUMBER OF VERTICES (INPUT)
C     IV - ARRAY OF MU'S(OUTPUT)
C
      DIMENSION IV(NN,NV)
      II=1
      MI=NV
      DO 30 I=1,NN
      IR=0
      IS=-1
      DO 20 J=1,MI
      DO 10 K=1,II
      IR=IR+1
      IV(I,IR)=IS
10    CONTINUE
      IS=-IS
20    CONTINUE
      II=II+II
      MI=MI/2
30    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ACFUNS(X,AF)
C
      DIMENSION X(15),AF(1)
      COMMON KODE,IFLAG,XITIME,ITRAP,JTRAP,ISENS,IPRIN,MKODE,EX,W,FR,NT,
     +NF,INDFF,NCO,IXX,MCODE,FF,IMA,IMA1,NOISE,IBANKF,IDCLOC
      COMMON/CO2/SEND(15),GR(15),ERR,NOPT,CONV,INEXP,IGR(15)
C
      DO 10 I=1,NOPT
      SEND(I)=ALOG(X(I))
10    CONTINUE
      CALL ANALYZ
C     WRITE(6,2000)KODE,(X(I),I=1,NOPT),ERR
2000  FORMAT(1X,"KODE=",I3,/,2X,6(E13.7,2X)/1X,"ERR=",F16.8)
      AF(1)=ERR
      RETURN
      END
```