REFERENCES

[1] S. Roberts and A. von Hippel, "A new method for measuring dielectric constant and loss in the range of centimeter waves," *J. Appl. Phys.*, vol. 17, pp. 610–616, 1946.
[2] W. B. Westphal, "Dielectric measuring techniques, A. Permittivity, 2. Distributed circuits," in *Dielectric Materials and Applications*, A. R. von Hippel, Ed. New York: Wiley 1954, ch. II, pp. 63–122.
[3] T. W. Dakin and C. N. Works, "Microwave dielectric measurements," *J. Appl. Phys.*, vol. 18, pp. 789–796, 1947.
[4] S. O. Nelson, C. W. Schlaphoff, and L. E. Stetson, "Computer program for calculating dielectric properties of low- or high-loss materials from short-circuited waveguide measurements," U. S. Dep. Agriculture, Agricultural Res. Service, Publ. No. ARS-NC-4, Nov. 1972.

The Complex Program for Constrained Minimization

PURPOSE: The complex program is a package of subroutines that minimizes a nonlinear objective function subject to nonlinear inequality constraints.

LANGUAGE: Fortran IV G Level 21 for the IBM 360/65 computer; 270 cards including comments but not including the user-supplied subprograms.

AUTHORS: K. F. Schenk, Department of Electrical Engineering, University of Ottawa, Ottawa, Ont., Canada.

AVAILABILITY: ASIS/NAPS Document No. 02272.

DESCRIPTION: The COMPLEX method [1]–[3] is a direct search procedure for the minimization of a nonlinear objective function subject to nonlinear inequality constraints. The problem under consideration can be mathematically described as follows:

minimize the objective function $f(x_1,x_2,\cdots,x_n)$
subject to the implicit constraints $g_j(x_1,x_2,\cdots,x_n) \leq 0, \quad j = 1,2,\cdots,m$
and the explicit constraints $x_i^l \leq x_i \leq x_i^h, \quad i = 1,2,\cdots,n.$

The method is based on the construction of a flexible figure of $v \geq (n + 1)$ vertices. This figure, or complex, can expand or contract in any direction and, at the same time, is made to satisfy all the constraints. A set of explicit constraints is required for each variable. These explicit constraints define a rectangular region within which the initial complex is generated about an initial feasible point. The additional $(v - 1)$ vertices are generated by the formula

$$x_i = x_i^l + r_i(x_i^h - x_i^l), \qquad i = 1,2,\cdots,n$$

in which the r_i are pseudorandom numbers over the interval 0 to 1.

The initial complex will be within the rectangular region of search, and the variables will be, therefore, automatically scaled. The implicit constraints may, however, not all be satisfied. If this is the case, those vertices which are outside the feasible region are moved halfway towards the centroid of the remaining vertices. This procedure is repeated for each vertex until all the implicit constraints are satisfied. The centroid is calculated from the expression

$$x_i^c = \frac{1}{v-1} \sum_{k=1,k\neq w}^{v} x_{i\,k}, \qquad i = 1,2,\cdots,n \tag{1}$$

in which $k = w$ corresponds to the worst vertex (largest function value).

After the initial complex is generated, the algorithm consists of the following steps.

Step 1: The objective function is calculated at each vertex and the one with the largest function value is rejected and replaced by another located a distance $\alpha(\alpha > 1)$ times as far from the weighted centroid of the remaining vertices. If this new vertex is feasible the process is repeated.

Step 2: If the new vertex generated in step 1 has the largest function value, then it is moved a distance $\beta(\beta < 1)$ times closer to the weighted centroid of the previous complex.

Step 3: If the new vertex generated in step 2 is also the worst, then a contraction of the whole complex is initiated about the best vertex so far obtained.

Step 4: If a vertex generated at any of the previous steps does not satisfy an implicit constraint, then that vertex is moved a distance $\beta(\beta < 1)$ times closer to the centroid of the previous complex. The process is repeated until the vertex enters the feasible region.

Step 5: If a vertex generated at any of the previous steps contains a variable which does not satisfy an explicit constraint, that variable is reset at a suitable distance (10^{-6}) inside the appropriate boundary.

The weighted centroid is calculated from the expression

$$x_i^c = \sum_{k=1,k\neq w}^{v} (f_k)^\gamma x_{i,k} \Big/ \sum_{k=1,k\neq w}^{v} (f_k)^\gamma, \qquad i = 1,2,\cdots,n \tag{2}$$

in which f_k is the value of the objective function which corresponds to the vertex x_k. The weighting factor γ may be chosen in the range $0 \leq \gamma \leq 2$. For $\gamma = 0$ (2) reduces to (1).

The iterative procedure is terminated when

$$\sigma = \left[\frac{1}{v} \sum_{k=1}^{v} (f_k - f_m)^2\right]^{1/2} \leq \varepsilon \tag{3}$$

in which σ is the standard deviation, f_m the mean of the function values at all the vertices, and ε a predetermined tolerance. For double precision arithmetic (12 significant digits for the IBM 360/65) ε may be chosen as low as 10^{-5}; otherwise a value of 10^{-2} is recommended.

The computer program may be called as follows:

```
CALL COMPLX(N,X,F,M,V,XL,XH,ALFA,BETA,GAMMA,SIGMA,EPS)
```

where N,X,F,M,V,XL,XH,ALFA,BETA,GAMMA,SIGMA,EPS correspond to $n,x_i,f(x_1,\cdots,x_n),m,v,x_i^l,x_i^h,\alpha,\beta,\gamma,\sigma,\varepsilon$, respectively, and $i=1,2,\cdots,n$.

It was convenient to place the following user-specified variables in COMMON/CMPLX/COUNT,EVAL,T,IT,LOOP,NLOOP,UNIT,IPRINT,IDATA in which

COUNT	integer variable specifying the number of iterations; initially set to zero;
EVAL	integer variable specifying the number of function evaluations; initially set to zero;
T,IT	integer variables (if T = 1 printing occurs every IT iterations; otherwise every IT function evaluations);
LOOP,NLOOP	integer variables for convergence criterion (the iterative procedure is tested every NLOOP iterations by the counter LOOP which is initially set to zero);
UNIT	integer variable specifying the output device;
IDATA	logical variable (if IDATA = .TRUE. the input data is printed out, otherwise not);
IPRINT	logical variable (if IPRINT = .TRUE. intermediate printing will occur, otherwise not).

All the parameters, except F, must be supplied in the main program by the user. The user must supply the tolerance ε and he has some control over the printout. The random numbers are generated internally. This is described in ASIS/NAPS Document No. 02272.

REQUIRED SUBPROGRAMS

SUBROUTINE FUNCT (N,X,F): This user-supplied program provides the value of the objective function F at point X.

SUBROUTINE MONIT (N,X,F,COUNT,EVAL,M,SIGMA,UNIT): This user-supplied program provides the variables in the argument list for printing. This program is used for the intermediate printing as well as for the final printing.

FUNCTION G(L,X,N): This user-supplied program provides the implicit constraints. Considering the case of two implicit constraints

Manuscript received March 19, 1973; revised August 7, 1973. This work was supported by the National Research Council of Canada under Grant A8837.

See NAPS Document No. 02272 for 30 pages of supplementary material. Order from ASIS/NAPS, c/o Microfiche Publications, 305 E. 46th St., New York, N. Y. 10017. Remit in advance for each NAPS accession number $1.50 for microfiche or $5.00 for photocopies. Make checks payable to Microfiche Publications.

of the form $g_1(x_1,x_2) = x_1^2 - x_2^2 \leq 0$ and $g_2(x_1,x_2) = x_1 + x_2 \leq 0$, the required statements (in double precision) are

```
  DOUBLE PRECISION FUNCTION G(L,X,N)
  DOUBLE PRECISION X
  DIMENSION X(2)
  GO TO (1,2),L
1 G = X(1)**2 - X(2)**2
  RETURN
2 G = X(1) + X(2)
  RETURN
  END
```

Comments

The package has been programmed to handle up to 10 variables and a complex of 21 vertices. These restrictions may be changed by the user. The input parameters α, β, γ, and v should be chosen within the ranges $1.3 \leq \alpha \leq 2.0$; $0.2 \leq \beta \leq 1.0$; $0 \leq \gamma \leq 2.0$; and $n + 1 \leq v \leq 2n + 1$. The program was run in double precision arithmetic and used 38K units of computer memory. This includes the numerical example provided with the ASIS/NAPS Document. The numerical example has been taken from [4]. The problem has five variables and six implicit constraints. It took 14.7 s of computer time to obtain the solution to a tolerance of $\epsilon = 10^{-4}$.

Acknowledgment

The author wishes to thank the Computing Centre of the University of Ottawa for the provision of computing and other facilities.

References

[1] M. J. Box, "A new method for constrained optimization and a comparison with other methods," *Comput. J.*, vol. 8, pp. 42–52, 1965.
[2] S. L. S. Jacobi, J. S. Kowalik, and J. T. Pisso, *Iterative Methods for Nonlinear Optimization Problems*. Englewood Cliffs, N. J.: Prentice-Hall, 1972, pp. 195–197.
[3] G. S. G. Beveridge and R. S. Schechter, *Optimization: Theory and Practice*. New York: McGraw-Hill, 1970, pp. 453–456.
[4] K. F. Schenk and P. J. Lawrenson, "A comparison of the complex method of optimization with the penalty function approach using Zangwill's method for constrained optimization problems," *SWIEECO Rec.*, pp. 580–588, 1973.

CANOPT—Cascaded Network Optimization Package

PURPOSE: The program analyzes and optimizes certain cascaded linear time-invariant networks in the frequency domain made up of two-port elements such as resistors, inductors, capacitors, lossless transmission lines, lossless short-circuited, and open-circuited transmission line stubs, series and parallel *RLC* resonant circuits and microwave allpass *C*- and *D*-sections.

LANGUAGE: Fortran IV; 1578 cards.

AUTHORS: J. W. Bandler and J. R. Popović, Communications Research Laboratory and Department of Electrical Engineering, McMaster University, Hamilton, Ontario, Canada, L8S 4L7.

AVAILABILITY: ASIS/NAPS Document No. 02300. User's manual with an example and program listing is also available from J. W. Bandler at $30.00. A copy of the source deck will be made available for $100.00.

DESCRIPTION: A companion paper [1] in this issue presents the theory and organization of the package. The user's manual referred to above [2] has details on how to use the program and an additional example.

The package was tested on a CDC 6400, and requires about $18K_{10}$ or around 45 000 words in octal.

References

[1] J. W. Bandler, J. R. Popović, and V. K. Jha, "Cascaded network optimization program," this issue, pp. 300–308.
[2] J. W. Bandler and J. R. Popović, "CANOPT-Cascaded network optimization package," McMaster Univ., Hamilton, Ont., Canada, Communications Res. Lab., Internal Rep. CRL-12, Nov. 1973.

Manuscript received November 9, 1973. This work was supported in part by the National Research Council of Canada under Grant A7239 and the Communications Research Laboratory, McMaster University, Hamilton, Ont., Canada.
See NAPS Document No. 02300 for 66 pages of supplementary material. Order from ASIS/NAPS, c/o Microfiche Publications, 305 E. 46th St., New York, N. Y. 10017. Remit in advance for each NAPS accession number $1.50 for microfiche or $10.40 for photocopies. Make checks payable to Microfiche Publications.