INTERNAL REPORTS IN

SIMULATION, OPTIMIZATION AND CONTROL

No. SOC-131

TEACHING OPTIMAL DESIGN

J. W. Bandler

September 1976

FACULTY OF ENGINEERING
McMASTER UNIVERSITY
HAMILTON, ONTARIO, CANADA

TEACHING OPTIMAL DESIGN

John W. Bandler, Senior Member, IEEE.

Abstract

Experiences and views related to teaching optimal design to electrical engineering undergraduates as well as course content are discussed in the context of numerical methods of analysis and design. A number of documented user-oriented computer programs extensively used by students in modeling and optimization of circuits and systems are referenced and are available from the author. Two of them, namely CANOP2 and MINOPT, are briefly described.

The author is with the Department of Electrical Engineering, McMaster University, Hamilton, Canada L8S 4L7.

I. INTRODUCTION

This paper presents some experiences and views on teaching engineering design via numerical optimization techniques [1,2] to electrical engineering undergraduates. The context of the material is computer aided circuit and system analysis and design taught through appropriate courses.

Numerical analysis has long been a respectable subject for electrical engineers. The advent of the modern, high-speed, large memory digital computer has, within the framework of analysis-based engineering courses, been used mainly to solve larger computational problems than before without a corresponding advance in design philosophy. The transition: slide rule to computer, graph paper to CRT display, has still not, in the author's opinion, had a significant impact on the educator's outlook in optimal design.

This paper indicates current possibilities and limitations in both the application of optimization techniques as tools in design as well as in the undergraduate classroom. Some available programs are described or referenced [3-9].

II. COURSE CONTENT

Background Material

In order to develop meaningful procedures of optimization and optimal design using computer aids, it is assumed that the electrical engineering undergraduate has already been exposed to the following topics: Matrix analysis of linear systems. Steady state and transient analysis using a digital computer. Sparse matrix techniques. Sensitivity and tolerance analysis. Computer solution of electromagnetic fields using iterative techniques. Nonlinear d.c. circuit analysis. It is also assumed that the student is already acquainted with least squares approximation, minimization by steepest descent and simple (nongradient) direct search methods. As a guiding light (for the instructor, at least) Calahan´s book [10] is still recommended. Chua and Lin [11] and Director [12] also cover much of the needed background.

Procedure

One aim of the course is to explain the underlying concepts of efficient iterative methods of solving constrained optimization problems, to indicate their limitations as well as their potential. Another is to present a variety of ways of formulating engineering design problems as optimization problems. Finally, all the ideas are brought together in hands on experience using any of several packaged batch or interactive optimization programs in the context of a design project individually tailored to the student´s interests and progress.

Tests and assignments are employed mainly to determine whether basic concepts have been grasped. The students build up towards the final project which dominates their time towards the end. The project report usually represents over 50 percent of the grade. There is no final examination.

Optimization and Optimal Design

Figure 1 illustrates the most essential concepts involved in optimization such as the objective function U of several variables $\underset{\sim}{\phi}$, the Taylor expansion assuming differentiability, the gradient vector operator $\underset{\sim}{\nabla}$ and the Newton-type iteration (jth step) which seeks a point satisfying (for a quadratic model) the necessary condition for optimality of a zero gradient vector. It is felt that the parallel development of the basic process of solving a system of nonlinear equations $\underset{\sim}{f}(\underset{\sim}{\phi})=\underset{\sim}{0}$ along with the one-dimensional examples aid the student's understanding.

Figure 2 summarizes typical engineering design situations treated in the course [1,2]. Fig. 2(a) depicts upper and lower specifications on a response function of an independent variable ψ (e.g., frequency or time) implying a constraint region in the $\underset{\sim}{\phi}$ space. Fig. 2(b) represents a Chebyshev or minimax approximation problem involving three extreme error functions e_1, e_2 and e_3 with the corresponding $\underset{\sim}{\phi}$-space representation of $\max_i |e_i|$. Depicted is the phenomenon of discontinuous derivatives occuring when the max function shifts from one error function to another. The situation of many circuits with independent design parameter values lying within a tolerance region of a nominal design [13] is depicted in Fig. 2(c). Here, a whole production line of designs may be involved.

Features of the nonlinear programming problem and nonlinear minimax approximation problem [2], which are central to optimal design, are sketched in parallel in Fig. 3. The objective functions are scalars. The constraints are explicit in the first problem and implied in the second. Necessary conditions for optimality involving nonegative multipliers u_i with corresponding diagrams are contrasted. The understanding of algorithms and the interpretation of solutions are crucially related to the optimality conditions and hence the author dwells on them with a variety of illustrations of different cases. One case is the unifying example 2 of Fig. 3. Here, we may interpret optimality either in the nonlinear programming or minimax senses.

Figure 4 presents two rather general approaches to solving constrained optimization problems. The first is the widely used Fiacco-McCormick barrier function method involving a sequence of unconstrained solutions converging to the desired solution from the interior of the constraint region [14]. The second is an example of an exact penalty function method where a sufficiently large value of α will make the unconstrained minimax solution the desired one [15]. In the latter case constraints do not have to be satisfied during optimization, but in the former case they do.

Figure 5 illustrates the basic approach to generalized least pth optimization when the maximum M of a set of functions is either positive or negative [16,17]. In the former case of Fig. 5(a) we see the normalization of the functions, the retention only of positively going functions followed by the formation of a scalar least pth objective equivalent to a penalty function. In the latter case of Fig. 5(b) the normalization of all the functions also changes their sign. All functions are retained in a barrier type objective function. Minimization of the least pth objective in an engineering design problem, therefore, tends to pull a response towards a specification if the specification is violated and increase the margin by which the specification is satisfied if subsequently possible.

One of the most exciting areas developed in the course is that of optimal centering, tolerancing and tuning [13]. It is a difficult problem to formulate efficiently in general and is still under intensive research. The results are extremely worthwhile in practice. Sufficiently straightforward examples such as resistive voltage divider circuits can be found to enable the students to program and solve meaningfully posed design problems.

Fig. 6 illustrates in one dimension the features of optimal worst case design. All possible production outcomes define a tolerance region, usually characterized by the nominal point ϕ^0, tolerance ε and parameter μ such that

$$\phi = \phi^0 + \varepsilon\mu , \quad -1 \leq \mu \leq 1$$

is an outcome. A tuned outcome is given by

$$\phi = \phi^0 + \varepsilon\mu + t\rho$$

$$-1 \leq \mu \leq 1, \quad -1 \leq \rho \leq 1$$

where t represents the range and ρ the setting of the control. Depending on whether the tolerance exceeds the tuning as in Fig. 6(a) or the tuning exceeds the tolerance as in Fig. 6(b) we obtain, respectively, an effective tolerance region which must be entirely contained in the constraint region for 100% yield or an effective tuning region only one point of which need be in the constraint region. Generalizing the basic statement of the centering-tolerancing-tuning problem to many dimensions is relatively simple, but may result in a vast nonlinear programming problem. The conventional assumption that the worst case can be predicted by linearizing the constraints at the nominal point is, within the scope of the course, probably as far as one can go in developing a computationally feasible formulation.

III. PROGRAM PACKAGES

Here, two of the packages available to students will be given detailed attention.

Interactive Cascaded Network Optimization Package [4,18].

The package called CANOP2 will analyze and optimize cascaded, linear, time-invariant networks in the frequency domain. It is based on CANOPT [19]. It plots responses and enforces equality on the variable parameters, if desired.

The program is organized in such a way that future additions or deletions of performance specifications, constraints, optimization methods and circuit elements are readily implemented. Presently, the network is assumed to be a cascade of two-port building blocks terminated in a unit normalized, frequency-independent resistance at the source and a user-specified frequency-independent resistance at the load.

A variety of two-port lumped and distributed elements such as resistors, inductors, capacitors, lossless transmission lines, lossless short-circuited and open-circuited transmission-line stubs, series and parallel LC and RLC resonant circuits and microwave allpass C- and D-sections can be handled. Upper and lower bounds on all relevant parameters can be specified by the user. A generalized least pth objective function or sequence of least pth objective functions incorporating simultaneously input reflection coefficient, insertion loss, relative group delay and parameter constraints (if any) are automatically created. Constraints are treated by the objective function in essentially the same way as the performance specifications [19]. To distinguish between the various responses or constraint functions a scheme for interval translation and introduction of artificial points has been developed. The Fletcher method of minimizing unconstrained functions of many variables [20] is available to the user. The package incorporates the adjoint network method of sensitivity evaluation [2].

If equality (symmetry) of some parameters can be predicted, symmetry may be forced throughout the optimization. Results may be automatically presented numerically and graphically and analysis of different responses may be performed at the user's discretion and a new optimization may be requested at different frequencies. A summary of the features and options available is given in Table I.

The package written in FORTRAN IV was originally developed for batch processing on a CDC 6400 computer and has now been largely extended for use on INTERCOM. The user may interact at many points with the program to change parameters, frequency range, types and options and to request plots. The interactive user enters his data in free format, and is not required to learn any special language. He responds to simple questions in a straightforward manner.

<u>A Sequential Least pth Optimization Program</u> [5][21]

MINOPT is a package of subroutines for solving design problems in which the objective is to best satisfy a given set of design specifications or constraints in the least pth or minimax sense. It assumes the availability of first partial derivatives of the functions concerned with respect to the design parameters. Essentially, a single least pth approximation can be done, or a sequence of least pth approximations with finite constant p can be carried out to produce highly accurate minimax soltuions, if desired. An estimated lower bound on the minimax solution is employed by the algorithm. A feature to successively drop functions likely to be inactive at the solution is incorporated. The program is efficient and well-suited to conducting feasibility checks.

MINOPT is written in FORTRAN IV and has been tested on a CDC 6400 computer. The following is a brief description of the subroutines called by MINOPT.

USER subprogram provided by the user to calculate the functions and first partial derivatives.

LPOBJ formulates the least pth objective.

GDCHK checks the derivatives at the starting point by numerical perturbation.

OUTPUT outputs the optimum solution or the current estimate of the solution.

VA09A is the Fletcher minimization program.

Consider finding a second-order model of a fourth-order system, when the input to the system is an impulse, in the minimax sense. The transfer function of the system is

$$G(s) = \frac{(s+4)}{(s+1)(s^2 + 4s + 8)(s+5)}$$

and of the model is

$$H(s) = \frac{\phi_3}{(s+\phi_1)^2 + \phi_2^2}$$

The problem is therefore equivalent to making the function

$$F(\underset{\sim}{\phi},t) = \frac{\phi_3}{\phi_2} \exp(-\phi_1 t) \sin \phi_2 t$$

best approximate

$$S(t) = \frac{3}{20} \exp(-t) + \frac{1}{52} \exp(-5t) - \frac{\exp(-2t)}{65} (3\sin 2t + 11\cos 2t)$$

in the minimax sense.

The problem was discretized in the time interval 0 to 10 seconds and the function to be minimized is

$$\max_{i \in I} |e_i(\underset{\sim}{\phi})| \qquad I = \{1, 2, \ldots, 51\}$$

where

$$e_i(\underset{\sim}{\phi}) = F(\underset{\sim}{\phi}, t_i) - S(t_i)$$

A printout of the results is shown in Fig. 7. Four optimizations and 119 funtion evaluations are required. It is interesting to observe the successive reduction in number of error functions actually calculated, so that the computing effort is far less than implied by the number 119.

IV. DISCUSSION

Documented, user-oriented computer programs extensively used by students in modeling and optimization of circuits and systems are available from the author. The examples solved by the students range over analog and digital circuits and systems, active and passive circuit design, low frequency and microwave design, frequency domain and time domain approximation, etc.

The efficient evaluation and utilization of sensitivities is central to the course. Not only nominal design may be achieved iteratively followed, for example, by tolerance analysis, but optimally centered, toleranced and tuned designs are possible with reasonable additional effort. Sufficiently straightforward examples can be found to demonstrate conclusively the value of the general approach as well as the justification in exposing these ideas to the undergraduate student.

The solution of differential equations, the solution of nonlinear equations and the minimization of nonlinear functions of many variables are crucial to engineering modeling, analysis and design. Once students have mastered the concepts of steepest descent and the basic Newton-type iteration, solution methods for these problems do not require much additional stretch of the imagination.

The author's preferred method of student evaluation in a design-oriented course is an individual project followed by a report. This allows students to move at different paces, permits experiences with different problems to be shared among them and encourages depth and breadth from the more motivated student. The main difficulty in managing this approach has been in convincing the students that they have insufficient time to complete projects large enough to satisfy their own ambitions.

It is stressed that the efficient utilization of algorithms for design requires a reorientation in the thinking of the engineer who may or may not be well-versed in simulation. Putting a simulation program into a simple loop, whether the designer is in that loop or not, severly limits his horizons. Nevertheless, a large number of simulation programs exist which do not provide for efficient means of changing design parameters as needed by design, for example, efficient first or large-change sensitivity evaluation is not provided for. The graduating engineer will likely meet many such programs. An approach to exploit such programs appropriately and with minimal effort involves multi-dimensional low-order approximations [22]. These approximations are also useful in modeling of experimental data and surface fitting in general.

The author believes that the student´s time is at a premium and that the material presented to him should be sufficiently fundamental to be of value during advances in hardware and software. It seems, therefore, preferable to avoid the use of large, general purpose simulation programs if: (a) they require considerable investment of the student´s time in mastering inessential details involved in their use at the expense of the theory of basic iterative methods, or (b) they detract from the process of acquiring the expertise of setting up and running design problems <u>automatically</u>.

Probably the most difficult and time-consuming topics for the student ot master are those of defining a design problem to the computer in terms of appropriate objectives, choice of non-redundant design variables, selection of essential performance and other constraints, proper scaling to facilitate rapid convergence and, above all, the correct interpretation of false or (for any reason) undesirable solutions. In contrast, subjects such as sensitivity evaluation and minimization algorithms (in their basic form) are readily grasped.

V. CONCLUSIONS

At a time of increased specialization, the optimization approach to problem solving is particularly opportune. It provides the engineer with a tool, e.g., a modest package for minimizing functions subject to constraints, which is applicable with varying effectiveness or efficiency to such diverse problems as transistor and other device modeling, rational function approximation, curve fitting, nonlinear (and linear) circuit analysis, tolerance assignment and post production tuning strategies. Branch and bound strategies, the essence of which are rather straightforward, permit discrete solutions to be forced, e.g., in digital filter design or in the optimal utilization of available components. The use of off-the-shelf components or the suitable restriction of design parameter values or the use of loosely toleranced elements obviously reduces the cost of production. This philosophy is stressed in the classroom, and the available programs can help in realizing these objectives.

Two volumes should finally be mentioned as providing excellent background and motivation for the instructor, namely the reprint volumes of Director [23] and Szentirmai [24].

REFERENCES

[1] J.W. Bandler, "Optimization methods for computer-aided design", IEEE Trans. Microwave Theory Tech., vol. MTT-17, Aug. 1969, pp. 533-552. Also in Microwave Integrated Circuits, J. Frey, Ed., Artech House, 1975.

[2] J.W. Bandler, "Computer-aided circuit optimization", in Modern Filter Theory and Design, G.C. Temes and S.K. Mitra, Eds., New York: Wiley - Interscience, 1973.

[3] J.H.K. Chen, "DISOPT - a general program for continuous and discrete non-linear programming problems", Faculty of Engineering, McMaster Univ., Hamilton, Canada, Report SOC-29, Mar. 1974. (Revised June 1975.)

[4] J.W. Bandler and J.R. Popovic, "CANOP2 - interactive cascaded network optimization package", Faculty of Engineering, McMaster Univ., Hamilton, Canada, Report SOC-69, Dec. 1974.

[5] J.W. Bandler, C. Charalambous and J.H.K. Chen, "MINOPT - an optimization program based on recent minimax results", Faculty of Engineering, McMaster Univ., Hamilton, Canada, Report SOC-70, Dec. 1974. Computer Program Description in IEEE Trans. Microwave Theory Tech., vol. MTT-24, Aug. 1976, p. 543.

[6] W.Y. Chu, "Extrapolation in least pth approximation and nonlinear programming", Faculty of Engineering,

McMaster Univ., Hamilton, Canada, Report SOC-71, Dec. 1974.

[7] J.W. Bandler and W.Y. Chu, "FLOPT2 - a program for least pth optimization with extrapolation to minimax solutions", Faculty of Engineering, McMaster Univ., Hamilton, Canada, Report SOC-84, Apr. 1975. Computer Program Description in IEEE Trans. Microwave Theory Tech., vol. MTT-24, Aug. 1976, pp. 543-544.

[8] J.W. Bandler, J.H.K. Chen, P. Dalsgaard and P.C. Liu, "TOLOPT - a program for optimal, continuous or discrete, design centering and tolerancing", Faculty of Engineering, McMaster Univ., Hamilton, Canada, Report SOC-105, Sept. 1975.

[9] J.W. Bandler and W.Y. Chu, "FLOPT3 - an interactive program for least pth optimization with extrapolation to minimax solutions", Faculty of Engineering, McMaster Univ., Hamilton, Canada, Report SOC-108, Oct. 1975.

[10] D.A. Calahan, Computer Aided Network Design (revised ed.). New York: McGraw-Hill, 1972.

[11] L.O. Chua and P.M. Lin, Computer-aided Analysis of Electronic Circuits. Englewood Cliffs, N.J.: Prentice-Hall, 1975.

[12] S.W. Director, Circuit Theory - A Computational Approach. New York: Wiley, 1975.

[13] J.W. Bandler, P.C. Liu and H. Tromp, "A nonlinear programming approach to optimal design centering, tolerancing and tuning", IEEE Trans. Circuits and Systems, vol.

CAS-23, March 1976, pp. 155-165.

[14] A.V. Fiacco and G.P. McCormick, Nonlinear Programming: Sequential Unconstrained Minimization Techniques. New York: Wiley, 1968.

[15] J.W. Bandler and C. Charalambous, "Nonlinear programming using minimax techniques", J. Optimization Theory and Applications, vol. 13, June 1974, pp. 607-619.

[16] J.W. Bandler and C. Charalambous, "Theory of generalized least pth approximation", IEEE Trans. Circuit Theory, vol. CT-19, May 1972, pp. 287-289. Also in Computer-Aided Filter Design, G. Szentirmai, Ed., IEEE Press, 1973.

[17] C. Charalambous, "A unified review of optimization", IEEE Trans. Microwave Theory Tech., vol. MTT-22, March 1974, pp. 289-300.

[18] J.W. Bandler and J.R. Popovic, "Interactive cascaded network optimization package", Proc. 18th Midwest Symp. on Circuits and Systems (Montreal, Canada, Aug. 1975), pp. 525-529.

[19] J.W. Bandler, J.R. Popovic and V.K. Jha, "Cascaded network optimization program", IEEE Trans. Microwave Theory Tech., vol. MTT-22, March 1974, pp. 300-308.

[20] R. Fletcher, "A new approach to variable metric algorithms", Computer J., vol. 13, Aug. 1970, pp. 317-322.

[21] J.W. Bandler, C. Charalambous and J.H.K. Chen, "MINOPT - an optimization program based on recent minimax results", Proc. 18th Midwest Symp. on Circuits and

Systems (Montreal, Canada, Aug. 1975), pp. 520-524.

[22] J.W. Bandler, H.L. Abdel-Malek, P.B. Johns and M.R.M. Rizk, "Optimal design via modeling and approximation", IEEE Int. Symp. Circuits and Systems (Munich, Germany, Apr. 1976), pp. 767-770.

[23] S.W. Director, Ed., Computer Aided Circuit Design: Simulation and Optimization. Stroudsburg, Pa.: Dowden, Hutchinson and Ross, 1973.

[24] G. Szentirmai, Ed., Computer-Aided Filter Design. New York: IEEE Press, 1973.

Figure Captions

Fig. 1 Contrast of essential concepts in minimization and solution of nonlinear equations, with one-dimensional illustrations.

Fig. 2 Typical engineering design situations.

(a) Upper and lower specifications with corresponding constraint region.

(b) Error function with contour of the maximum.

(c) Toleranced design satisfying the specifications.

Fig. 3 Contrast of essential concepts in minimization subject to constraints and minimax approximation, with two-dimensional and one-dimensional illustrations.

Fig. 4 Contrast of common barrier function approach and exact penalty function (minimax) approach to constrained minimization. One-dimensional illustrations show the effect of changing the parameters r and α .

Fig. 5 Illustration of generalized least pth optimization [16,17].

Fig. 6 One-dimensional illustration of the concepts involved in centering, tolerancing and tuning.

(a) Tolerance exceeds tuning.

(b) Tuning exceeds tolerance.

Fig. 7 Results for the system modelling example with $p=2$. Starting point $\underset{\sim}{\phi} = [1\ 1\ 1]^T$.

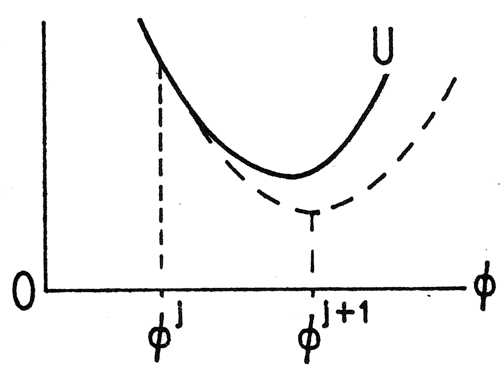
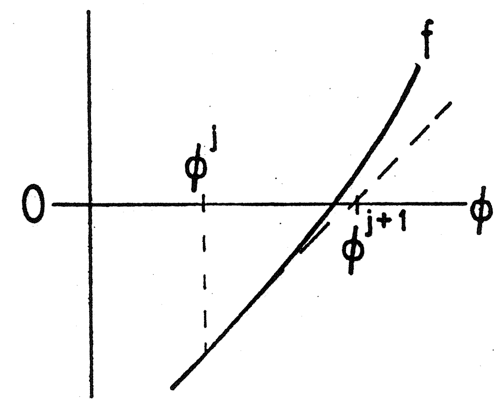
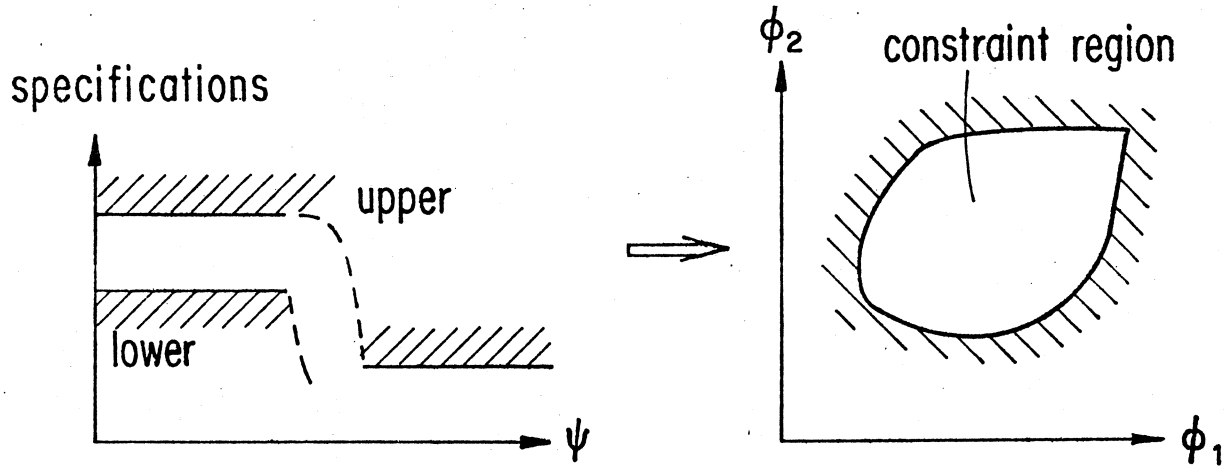
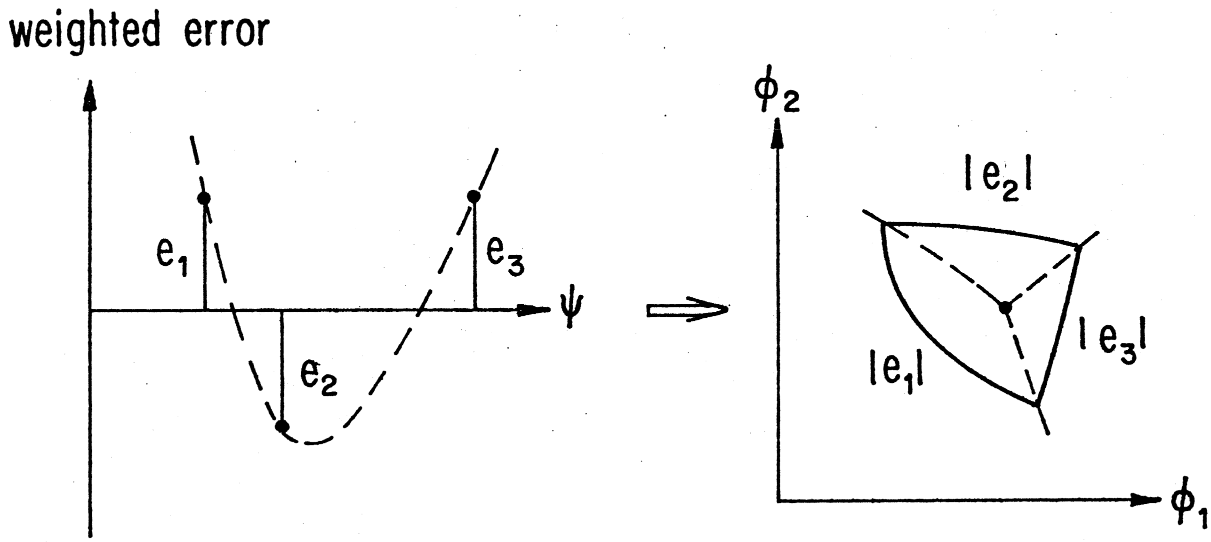
objective	$\min_{\phi} U(\phi)$	$\text{solve}_{\phi} \quad f(\phi)=0$
Taylor series	$U(\phi+\Delta\phi) = U + \nabla U^T \Delta\phi + 0.5\, \Delta\phi^T H \Delta\phi + \cdots$	
gradient	$\nabla U(\phi+\Delta\phi) = \nabla U + H\Delta\phi + \cdots$	$f(\phi+\Delta\phi) = f + J\Delta\phi + \cdots$
optimality	$0 = \nabla U + H\Delta\phi + \cdots$	$0 = f + J\Delta\phi + \cdots$
Newton	$H^j \Delta\phi^j = -\nabla U^j$	$J^j \Delta\phi^j = -f^j$
example		
	$\delta^j = -U'(\phi^j) / U''(\phi^j)$	$\delta^j = -f(\phi^j) / f'(\phi^j)$

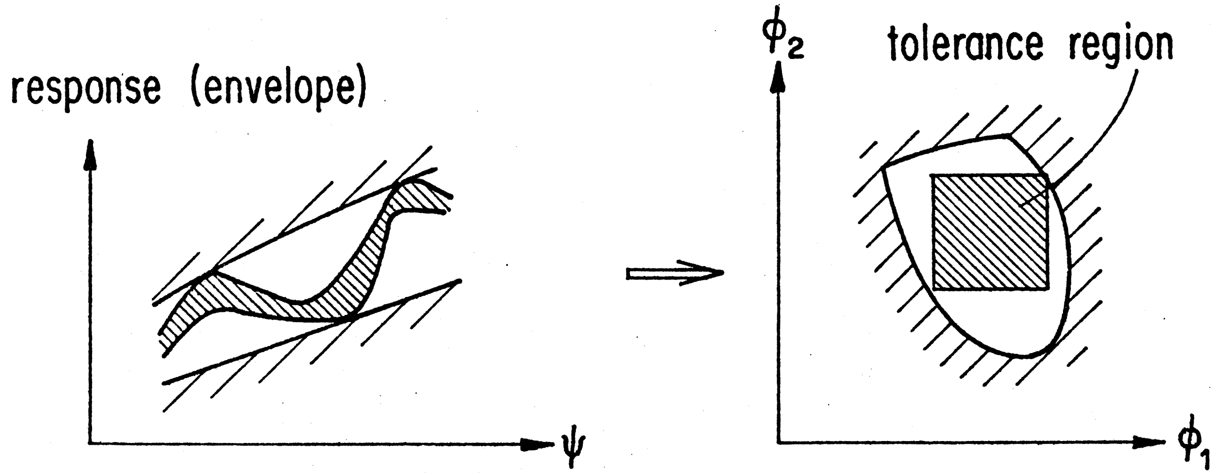
Fig. 1


specifications
upper
lower
ψ
ϕ_2
constraint region
ϕ_1

(a)


weighted error
e_1
e_2
e_3
ψ
ϕ_2
$|e_2|$
$|e_1|$
$|e_3|$
ϕ_1

(b)

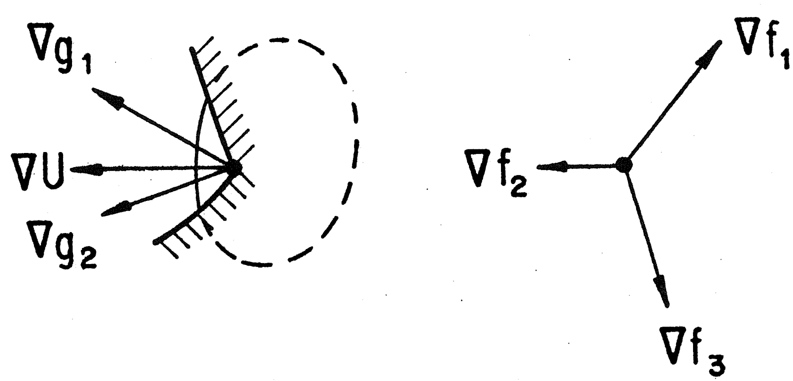

response (envelope)
ψ
ϕ_2
tolerance region
ϕ_1

(c)

Fig. 2

objective	$\min U$	$\min M$
function	$U = U(\phi)$	$M = \max_i f_i(\phi)$
constraints	$g_i(\phi) \geq 0$	$[M \geq f_i(\phi)]$
optimality	$\nabla U = \sum_i u_i \nabla g_i$	$0 = \sum_i u_i \nabla f_i$ $1 = \sum_i u_i$
multipliers	$u_i g_i = 0$ $u_i \geq 0$	$u_i (M - f_i) = 0$ $u_i \geq 0$

example 1



example 2

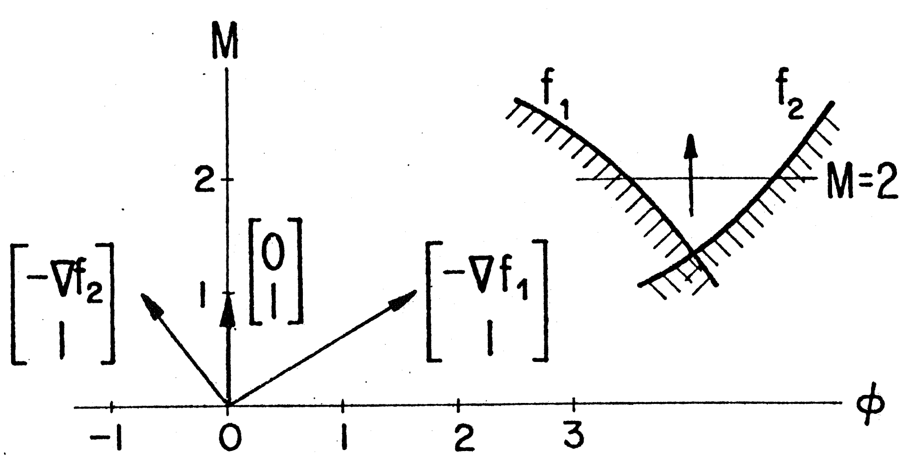


Fig. 3

objective	$\min_{\phi} B$	$\min_{\phi} V$
function	$B = U + r\sum_i 1/g_i$	$V = \max_i \left[U, U - \alpha g_i\right]$
constraints	$\left[g_i \geq 0\right]$	none
sequence	$0 < r^{j+1} < r^j$	$\alpha^{j+1} > \alpha^j > 0$
optimality	$0 = \nabla U - r\sum_i \frac{1}{g_i^2} \nabla g_i$	$0 = \nabla U - \sum_i v_i \alpha \nabla g_i$
multipliers	$u_i = \frac{r}{g_i^2}$	$v_i = \frac{u_i}{\alpha}$ $\sum_i v_i < 1$

example

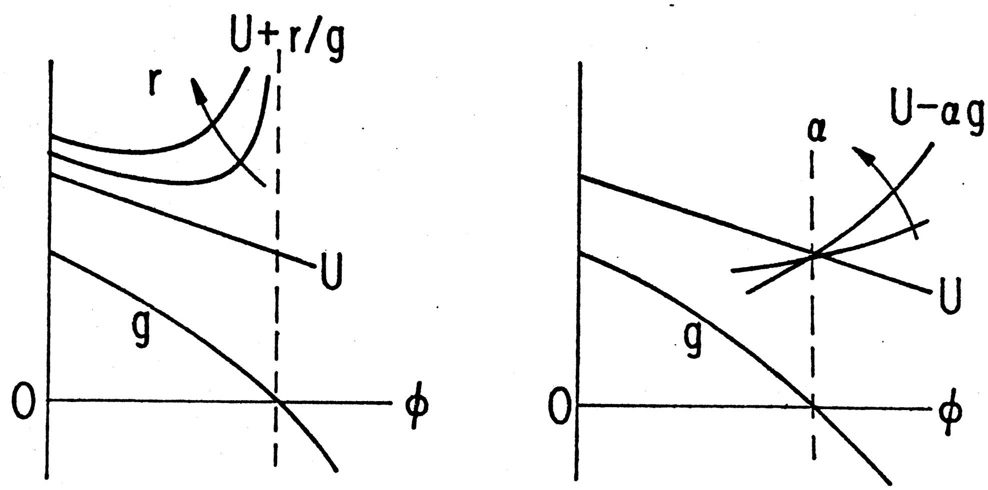


Fig. 4

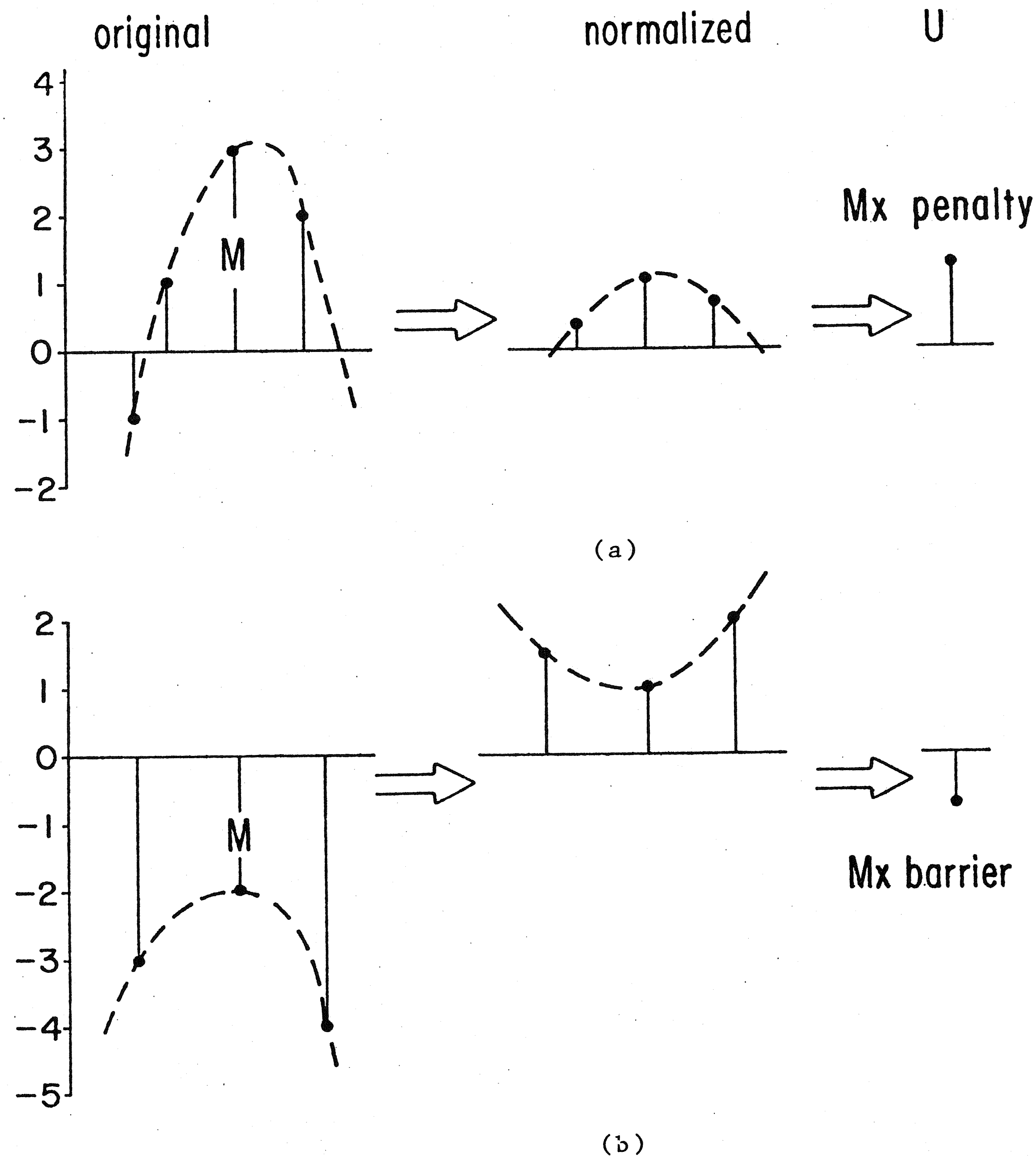
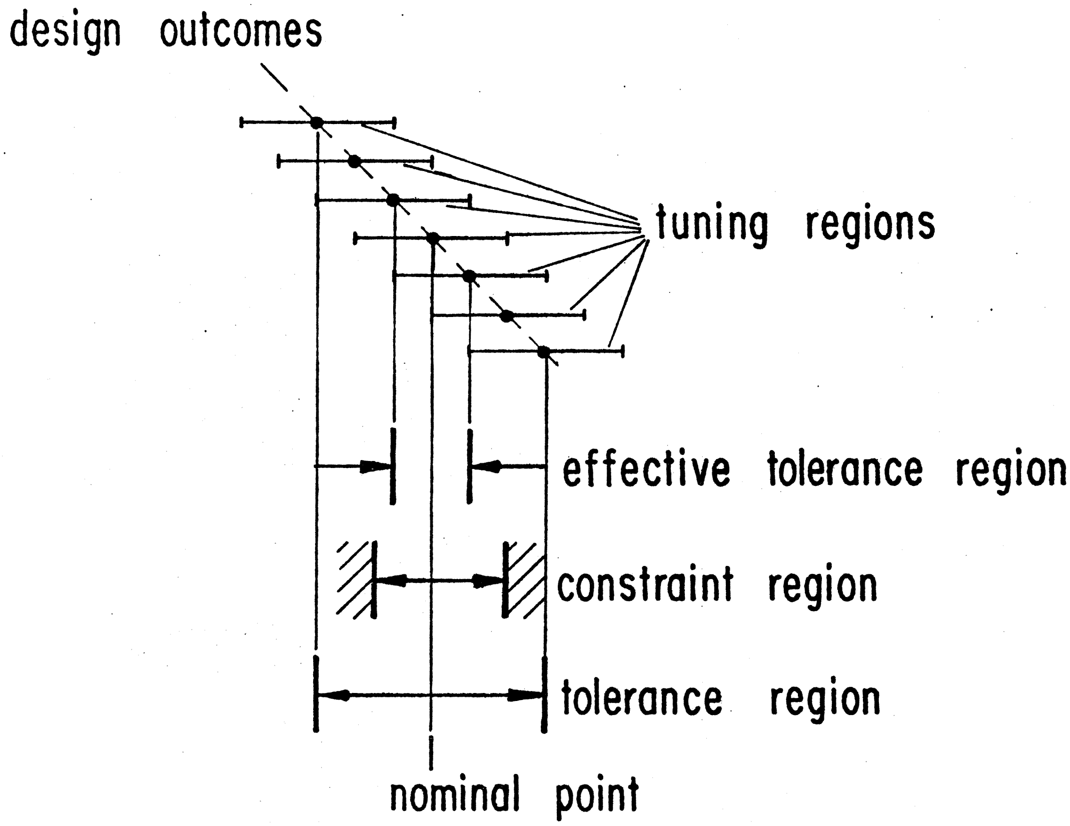
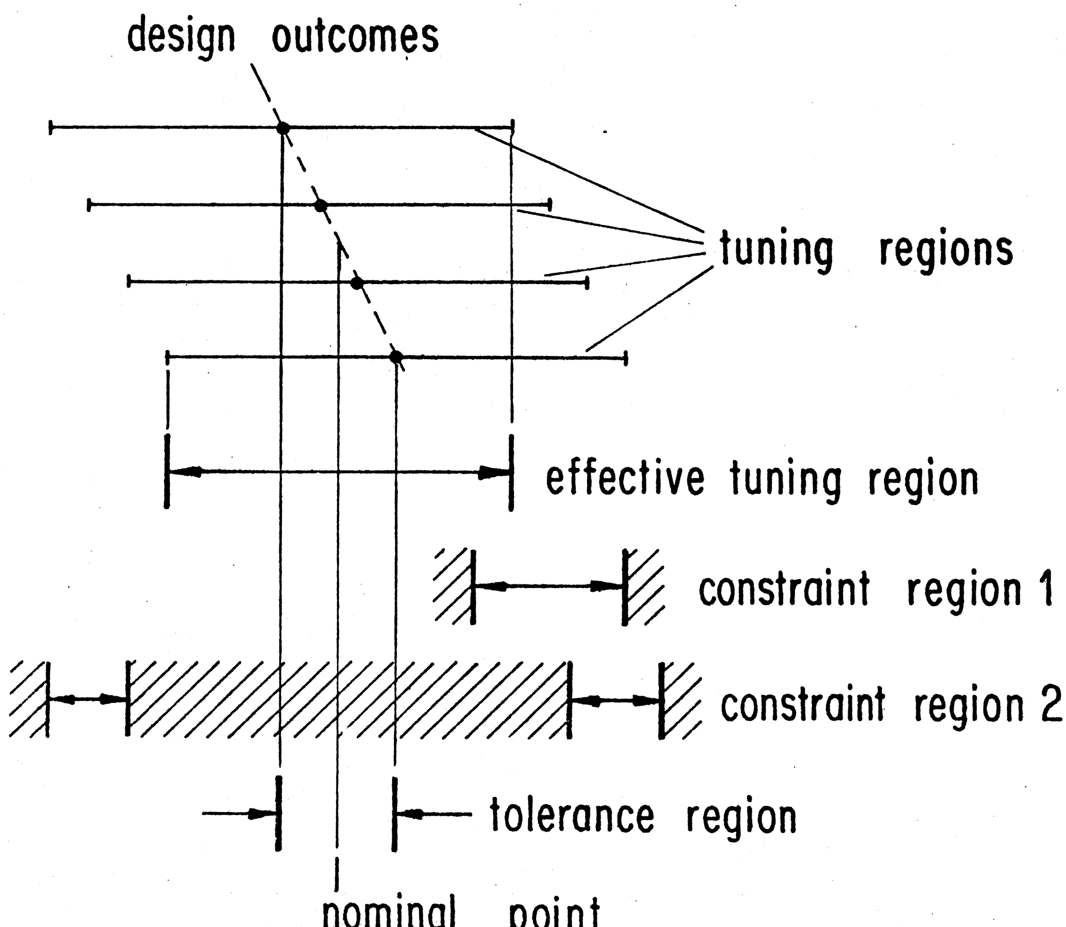

original
normalized
U
4
3
2
1
0
-1
-2
M
Mx penalty
(a)
2
1
0
-1
-2
-3
-4
-5
M
Mx barrier
(b)

Fig. 5


design outcomes
tuning regions
effective tolerance region
constraint region
tolerance region
nominal point

(a)


design outcomes
tuning regions
effective tuning region
constraint region 1
constraint region 2
tolerance region
nominal point

(b)

Fig. 6

TABLE 1

SUMMARY OF FEATURES, OPTIONS AND PARAMETERS REQUIRED

Features	Type	Options	Parameters
Objective Functions	Least pth	$1<p<\infty$	Value of p for each of a specified number of optimizations Artificial margin Difference in objective functions for termination
Performance Specifications and Parameter Constraints	Upper (+1.) Lower (-1.) Single (0.)	Reflection coefficient (1) Insertion loss (2) Group delay (3) Parameter value (0)	Normalization frequency Number of points and constraints / Number of bands or intervals For each: Specification/constraint Weighting factor Type Option Frequency (sample point) or parameter / Lower and upper frequencies (band edges) Number of subintervals
Analysis Optimization	 Gradient	Analysis only (0) Fletcher optimization method (1)	Option Specified or default values for: Number of iterations allowed Estimate of lower bound on objective function Test quantities for termination
Circuit Elements	Cascaded Two-port	Typical plus C- and D-sections	Number of elements Sequence of code numbers Parameter values Indicator for fixed, variable or equal (symmetrical) parameters Load resistance Parameters for C- and D-sections
Graph	Frequency response	Given response Other response Any range Automatic scaling Specified scaling	As many plots as desired Option Frequency (sample point) / Lower and upper frequencies (band edges)

ARTIFICIAL MARGIN FOR THE NEXT OPTIMIZATION = 4.00000000E-03
NUMBER OF FUNCTIONS FOR THE NEXT OPTIMIZATION = 51

OPTIMIZATION 1

ITER	FUNCT	OBJECTIVE	VARIABLE	GRADIENT
0	9	6.394211E-01	1.000000E+00	-7.787849E-01
			1.000000E+00	-3.780300E-01
			1.000000E+00	7.898472E-01
20	36	7.778212E-03	8.520020E-01	6.395533E-06
			8.935317E-01	1.384626E-05
			1.422568E-01	-8.362492E-05
22	38	7.778211E-03	8.520350E-01	-9.730915E-08
			8.935018E-01	-2.855661E-08
			1.422609E-01	6.064313E-07

IEXIT = 1
NORMAL EXIT
CURRENT MAXIMUM FUNCTION VALUE = 1.05144148E-02
ARTIFICIAL MARGIN FOR THE NEXT OPTIMIZATION = 7.27711352E-03
NUMBER OF FUNCTIONS FOR THE NEXT OPTIMIZATION = 13

OPTIMIZATION 2

ITER	FUNCT	OBJECTIVE	VARIABLE	GRADIENT
35	55	1.161221E-03	7.061282E-01	-1.504854E-08
			9.479483E-01	-3.011574E-08
			1.251141E-01	1.234177E-08

IEXIT = 1
NORMAL EXIT
CURRENT MAXIMUM FUNCTION VALUE = 8.24480216E-03
AFTIFICIAL MARGIN FOR THE NEXT OPTIMIZATION = 7.93591219E-03
NUMBER OF FUNCTIONS FOR THE NEXT OPTIMIZATION = 6

OPTIMIZATION 3

ITER	FUNCT	OBJECTIVE	VARIABLE	GRADIENT
40	68	5.436247E-05	6.876561E-01	-4.268539E-03
			9.525845E-01	-2.717345E-04
			1.231909E-01	1.732489E-01
49	81	1.915435E-05	6.847436E-01	-3.683231E-07
			9.540264E-01	1.349722E-07
			1.228994E-01	1.685067E-06

IEXIT = 1
NORMAL EXIT
CURRENT MAXIMUM FUNCTION VALUE = 7.95178792E-03
ARTIFICIAL MARGIN FOR THE NEXT OPTIMIZATION = 7.94705799E-03
NUMBER OF FUNCTIONS FOR THE NEXT OPTIMIZATION = 4

OPTIMIZATION 4

ITER	FUNCT	OBJECTIVE	VARIABLE	GRADIENT
60	111	3.631441E-09	6.844180E-01	-1.622166E-02
			9.540929E-01	-1.916376E-02
			1.228643E-01	1.199815E-01
64	118	1.629539E-09	6.844178E-01	-1.428569E-02

9.540931E-01 -3.729895E-03
1.228642E-01 1.650471E-01

IEXIT = 1
NORMAL EXIT
CURRENT MAXIMUM FUNCTION VALUE = 7.94705954E-03
ARTIFICIAL MARGIN FOR THE NEXT OPTIMIZATION = 7.94705910E-03

FOLLOWING IS THE OPTIMUM SOLUTION

OBJECTIVE FUNCTION U = 1.62953865E-09
X(1) = 6.84417768E-01 GU(1) =-1.42856936E-02
X(2) = 9.54093084E-01 GU(2) =-3.72989486E-03
X(3) = 1.22864249E-01 GU(3) = 1.65047054E-01

NUMBER OF FUNCTION EVALUATIONS = 119*

*This total includes the number of function evaluations required for gradient checking, minimization and the determination of the artificial margin and index set.

Fig. 7. Results for the system modelling example with p=2. Starting point $\underset{\sim}{\phi} = [1\ 1\ 1]^T$.

SOC-131

TEACHING OPTIMAL DESIGN

J.W. Bandler

September 1976, No. of Pages: 28

Revised:

Key Words: Optimal design, optimization programs, engineering education

Abstract: Experiences and views related to teaching optimal design to electrical engineering undergraduates as well as course content are discussed in the context of numerical methods of analysis and design. A number of documented user-oriented computer programs extensively used by students in modeling and optimization of circuits and systems are referenced and are available from the author. Two of them, namely CANOP2 and MINOPT, are briefly described.

Description: Published in IEEE Trans. Education, vol. E-20, Feb. 1977.

Related Work: SOC-69, SOC-70, SOC-113.

Price: $ 5.00.