Reprinted from JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS Vol. 14, No. 1, July 1974
Printed in Belgium

Optimization of Design Tolerances Using Nonlinear Programming[1]

J. W. BANDLER[2]

Communicated by C. T. Leondes

Abstract. A possible mathematical formulation of the practical problem of computer-aided design of electrical circuits (for example) and systems and engineering designs in general, subject to tolerances on k independent parameters, is proposed. An automated scheme is suggested, starting from arbitrary initial acceptable or unacceptable designs and culminating in designs which, under reasonable restrictions, are acceptable in the worst-case sense. It is proved, in particular, that, if the region of points in the parameter space for which designs are both feasible and acceptable satisfies a certain condition (less restrictive than convexity), then no more than 2^k points, the vertices of the tolerance region, need to be considered during optimization.

Key Words. Engineering design, nonlinear programming, convex programming, optimization theorems, approximation of functions.

1. Introduction

An extremely important practical problem is the problem of optimal design subject to tolerances. Recently published work (Refs. 1 6) has yielded some practical insight into the nature of the problem. Indeed, it

[1] This paper was presented at the 6th Annual Princeton Conference on Information Sciences and Systems, Princeton, New Jersey, 1972. The author has benefitted from practical discussions with J. F. Pinel and K. A. Roberts of Bell-Northern Research. V. K. Jha programmed some numerical examples connected with this work. C. Charalambous, P. C. Liu, and N. D. Markettos have made helpful suggestions. The work was supported by Grant No. A-7239 from the National Research Council of Canada.

[2] Associate Professor of Electrical Engineering, Department of Electrical Engineering, McMaster University, Hamilton, Ontario, Canada.

suggests immediately the possibility of formulating the complete worst-case design of circuits or systems as a nonlinear programming problem.

An automated scheme would start from an arbitrary acceptable or unacceptable design and, under appropriate restrictions, stop at an acceptable design which is optimum in the worst-case sense for specified tolerances. The most suitable objective function to be minimized would also seem to be one that best describes the cost of fabrication of the circuit or system, as suggested by some authors (Refs. 1–6).

It is the purpose of this paper to propose possible formulations and to discuss this problem generally. It is not claimed that a complete solution has been obtained. However, a number of interesting objective functions (more appropriately, perhaps, cost functions) have been investigated.

Many types of objective functions can be formulated. A number of variations on the sum of the inverses of the absolute tolerances or the sum of the inverses of the tolerances relative to the respective nominal parameter values can be obtained. Furthermore, the nominal parameter values may or may not be variable. The relative merits of these and other functions which attempt in some way to maximize the size of the region of possible designs (namely, the tolerance region) are discussed.

For the purposes of this paper, it is assumed that the parameter tolerances can be specified independently. Furthermore, it is assumed that the design parameters and tolerances can be varied continuously. The tolerance region, in this case, will be defined by simple upper and lower bounds on the parameters. Of course, the region will contain an infinite number of acceptable designs, assuming that it is a subregion of the intersection of regions of acceptable and feasible designs. It is proved that, if this region satisfies a certain condition (less restrictive than convexity), then only the (finite) number of vertices of the tolerance region need, at most, to be investigated.

2. Feasible and Acceptable Designs

A wide range of design problems can be formulated as nonlinear programming problems. One usually defines a scalar objective function $U(\phi)$, where

$$\phi \triangleq \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_k \end{bmatrix} \tag{1}$$

represents the k independent design parameters. Design constraints can be assembled into a column vector $g(\phi)$ and the problem can be stated as finding $\check{\phi}$ such that

$$U(\check{\phi}) = \min_{\phi \in R_c} U(\phi), \tag{2}$$

where

$$R_c \triangleq \{\phi \mid g(\phi) \geqslant 0\}. \tag{3}$$

For the purposes of the present discussion, let us assume that two kinds of constraint functions are present, those that determine the *feasibility* of a design [designated $g_f(\phi)$] and those that determine the *acceptability* of a design [designated $g_a(\phi)$]. Therefore, we will define a feasible region of points R_f as

$$R_f \triangleq \{\phi \mid g_f \geqslant 0\} \tag{4}$$

and an acceptable region of points R_a as

$$R_a \triangleq \{\phi \mid g_a \geqslant 0\}. \tag{5}$$

Thus, $R_c = R_f \cap R_a$. It is assumed that all sets are nonempty. Note that R_a is not necessarily a subset of R_f.

The objective function is usually set up so that a feasible solution is obtained at an interior point of the acceptable region and as far as possible (in some sense) from its boundary. The reasoning behind this is the hope that, when the design is fabricated, inevitable errors in the design parameters might yield, nevertheless, an acceptable design. It is this flexibility which can be exploited in the optimization of tolerances. Often,

$$U(\phi) = -\min_{i \in I_a} g_i(\phi), \tag{6}$$

where the index set I_a relates to constraints defining R_a. It follows then that

$$R_a = \{\phi \mid U(\phi) \leqslant 0\}. \tag{7}$$

3. Tolerance Region

Given a *nominal* point ϕ^0 and a set of nonnegative *tolerances* ϵ, where

$$\epsilon \triangleq \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_k \end{bmatrix} \geqslant 0, \tag{8}$$

we can define a region of *possible* designs R_t as

$$R_t \triangleq \{\phi \mid \phi_i^0 - \epsilon_i \leqslant \phi_i \leqslant \phi_i^0 + \epsilon_i\,, \qquad i = 1, 2,\ldots, k\} \tag{9}$$

or, equivalently,

$$R_t \triangleq \{\phi \mid \phi_i = \phi_i^0 + t_i\epsilon_i\,, -1 \leqslant t_i \leqslant 1, i = 1, 2,\ldots, k\}. \tag{10}$$

Obviously, depending on the location of ϕ^0 and the value of ϵ, R_t may or may not be a subset of R_c.

The tolerance problem is beginning to take shape: R_t should be placed inside R_c in some optimal manner by adjusting ϕ^0 and ϵ to optimal values $\check{\phi}^0$ and $\check{\epsilon}$. A serious development, however, is that all points $\phi \in R_t$ must satisfy $g \geqslant 0$. We have, effectively, to deal with an infinite number of constraints.

For any given point ϕ^0, we can view the functions $g(\phi)$ with respect to ϵ as follows. We let the origin of the ϵ-space correspond to ϕ^0 (transla-

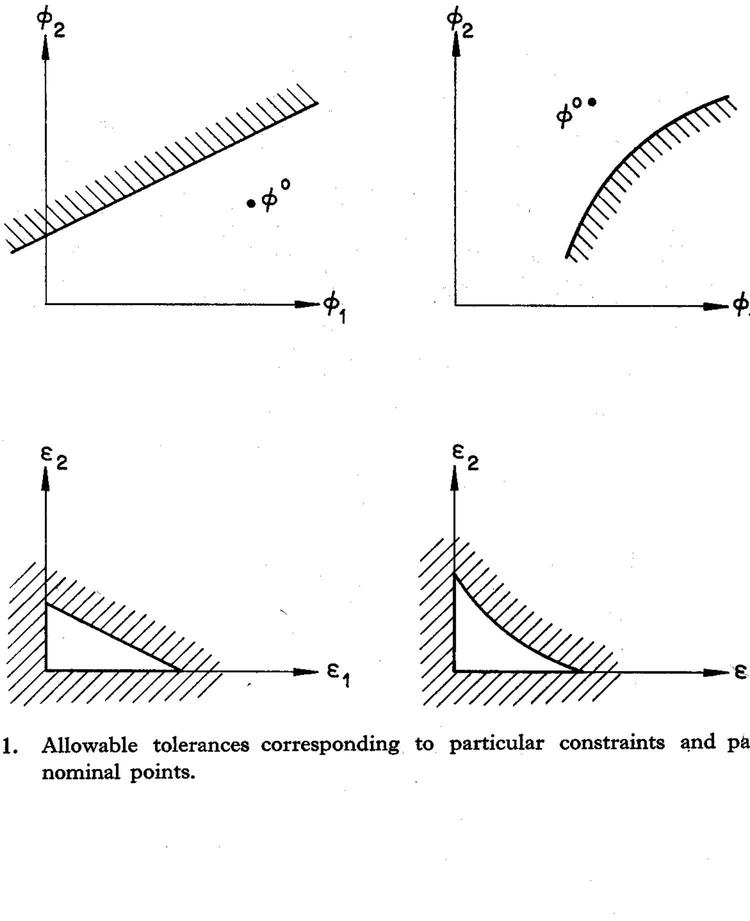

Fig. 1. Allowable tolerances corresponding to particular constraints and particular nominal points.

tion). Then, we consider all the possible linear parameter transformations [from (10)]

$$\epsilon = T(\phi - \phi^0)$$

suggested by the transformation matrix (magnification and reflection)

$$T \triangleq \begin{bmatrix} 1/t_1 & 0 & \cdots & 0 \\ 0 & 1/t_2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1/t_k \end{bmatrix}, \qquad -1 \leqslant t_i \leqslant 1, \quad i = 1, 2,..., k. \tag{11}$$

Two-dimensional examples of allowable tolerances in the tolerance space corresponding to particular constraints and particular nominal points in the parameter space are shown in Figs. 1–2.

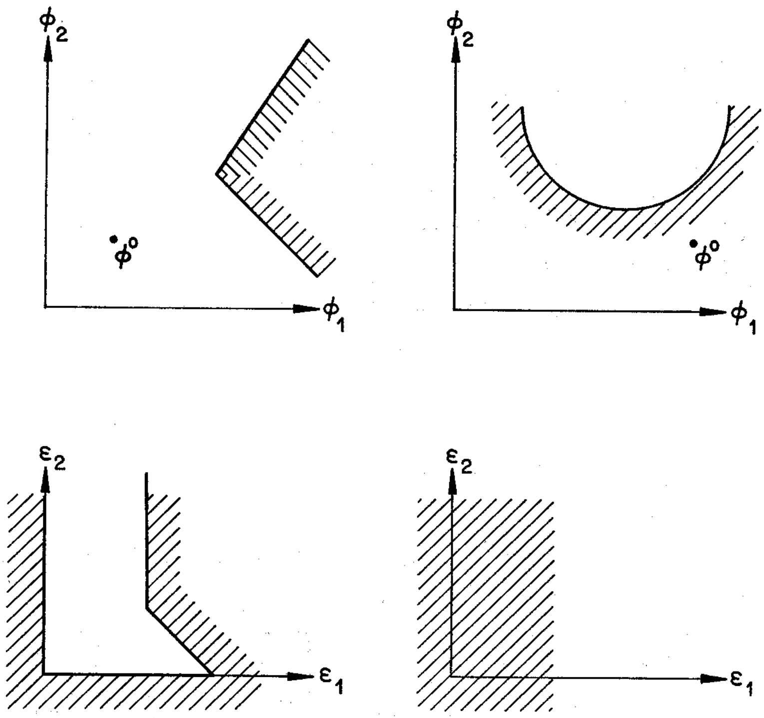

Fig. 2. Allowable tolerances corresponding to particular constraints and particular nominal points.

4. Restrictions on R_c

For obvious reasons, it is impractical to consider an infinite number of constraints. In order to make the problem tractable, a number of simplifying assumptions could be made to try to obtain a solution to the problem with reasonable computational effort.

It can be shown that, if R_c is convex, then from Refs. 7 or 8,

$$\phi^i \in R_c, \qquad i = 1, 2, \ldots, n. \tag{12}$$

implies that

$$\phi = \sum_{i=1}^{n} \lambda_i \phi^i \in R_c \tag{13}$$

for all λ_i satisfying

$$\sum_{i=1}^{n} \lambda_i = 1 \qquad \text{and} \qquad \lambda_i \geqslant 0, \quad i = 1, 2, \ldots, n. \tag{14}$$

For example, given a finite number of points ϕ^i in a finite-dimensional Euclidean space, it is easy to visualize that the ϕ^i are vertices of a polytope (the intersection of a finite number of closed halfspaces) and that ϕ is any interior or boundary point. If R_c is itself a polytope (all constraints linear), it is clearly convex.

The polytope R_t has 2^k vertices. Let the ith vertex be denoted by ϕ^i and let

$$\phi^i = \phi^0 - \epsilon + 2Ev_{i-1} \in R_c, \qquad i = 1, 2, \ldots, 2^k, \tag{15}$$

where

$$E \triangleq \begin{bmatrix} \epsilon_1 & 0 & \cdots & 0 \\ 0 & \epsilon_2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \epsilon_k \end{bmatrix} \tag{16}$$

and where v_i is a k-element vector whose elements reflect the subscript i in binary notation, i.e.,

$$v_0 \triangleq \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad v_1 \triangleq \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad v_2 \triangleq \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad v_3 \triangleq \begin{bmatrix} 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \ldots. \tag{17}$$

The vector v_{i-1} can be formed as follows:

$$v_{i-1} = \sum_{j=1}^{k} \mu_j(i)\, u_j \,, \tag{18}$$

where

$$\mu_1 \,, \mu_2 \,, \ldots, \mu_k \in \{0, 1\} \tag{19}$$

must satisfy (see Table 1)

$$i = 1 + \sum_{j=1}^{k} \mu_j(i)\, 2^{j-1}, \tag{20}$$

and where the k-element vectors u_j are given by

$$u_1 \triangleq \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad u_2 \triangleq \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \ldots, u_k \triangleq \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}. \tag{21}$$

Figure 3 illustrates an example in three dimensions. Observe that

$$Ev_{i-1} = \sum_{j=1}^{k} \mu_j(i)\, \epsilon_j u_j \,. \tag{22}$$

Table 1. Numbering scheme for the vertices of R_t.

i	$\mu_1(i)$	$\mu_2(i)$	$\mu_3(i)$	$\cdots$	$\mu_k(i)$	$\sum_{j=1}^{k} \mu_j(i)\, \epsilon_j u_j$
1	0	0	0		0	0
2	1	0	0		0	$\epsilon_1 u_1$
3	0	1	0		0	$\epsilon_2 u_2$
4	1	1	0		0	$\epsilon_1 u_1 + \epsilon_2 u_2$
5	0	0	1		0	$\epsilon_3 u_3$
6	1	0	1		0	$\epsilon_1 u_1 + \epsilon_3 u_3$
7	0	1	1		0	$\epsilon_2 u_2 + \epsilon_3 u_3$
8	1	1	1		0	$\epsilon_1 u_1 + \epsilon_2 u_2 + \epsilon_3 u_3$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
2^k	1	1	1		1	ϵ

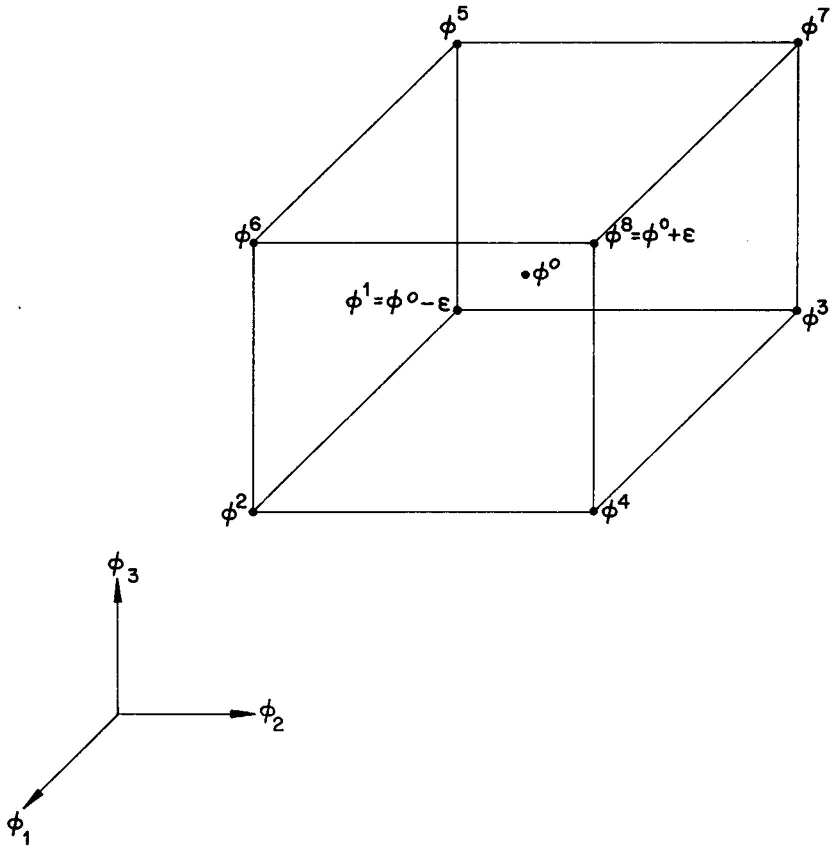

Fig. 3. Three-dimensional example of points defining the vertices of R_t.

Using (12)–(14), we have

$$\phi = \phi^0 - \epsilon + 2 \sum_{i=1}^{2^k} \left(\lambda_i \sum_{j=1}^{k} \mu_j(i)\, \epsilon_j u_j\right) \in R_c \tag{23}$$

if R_c is convex and the vertices of R_t are elements of R_c. Equation (23) generates the set R_t. Therefore, $R_t \subset R_c$. See Fig. 4.

It will now be shown that the assumption that R_c is convex is unnecessarily restrictive.

Theorem 4.1. If the vertices of R_t are in R_c, then $R_t \subset R_c$ if, for all $j = 1, 2,..., k$,

$$\phi^a, \phi^{b(j)} = \phi^a + \alpha u_j \in R_c , \tag{24}$$

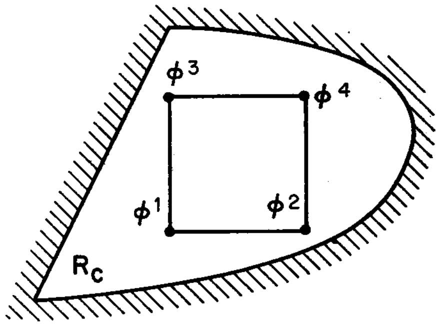

Fig. 4. Possible region R_c.

where α is a scalar, implies that

$$\phi = \phi^a + \lambda(\phi^{b(j)} - \phi^a) \in R_c \tag{25}$$

for all λ satisfying

$$0 \leqslant \lambda \leqslant 1. \tag{26}$$

See, for example, Fig. 5.

Proof. Let ϕ_l denote some point, in general, in an l-dimensional linear manifold generated by the first 2^l vertices as

$$\phi_l = \phi^0 - \epsilon + 2 \sum_{i=1}^{2^l} \left(p_i \sum_{j=1}^{l} \mu_j(i)\, \epsilon_j u_j \right), \tag{27}$$

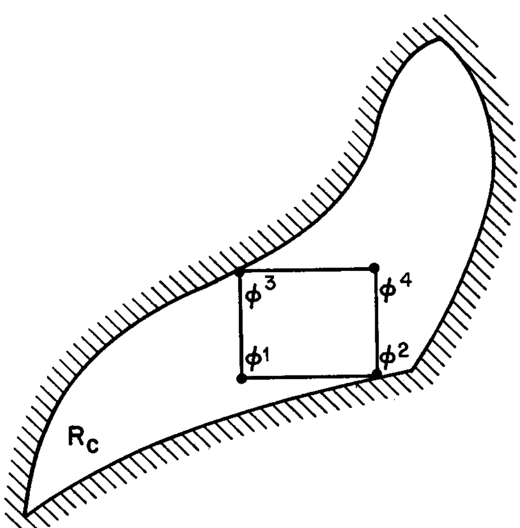

Fig. 5. Possible region R_c.

with p_i satisfying

$$\sum_{i=1}^{2^l} p_i = 1 \qquad \text{and} \qquad p_i \geqslant 0, \quad i = 1, 2,..., 2^l. \tag{28}$$

Note that, since max $i = 2^l$, we can deduce from (20) that

$$\mu_j = 0, \qquad j \geqslant l, \tag{29}$$

in (22), so that the relevant summation need be taken only up to l and not k.

Assume that $\phi_l \in R_c$ for all $\phi^i \in R_o$ given in (22). Now, consider

$$\phi_{l+1} = \phi^0 - \epsilon + 2 \sum_{i=1}^{2^{l+1}} \left(q_i \sum_{j=1}^{l+1} \mu_j(i)\, \epsilon_j u_j \right), \tag{30}$$

with q_i satisfying

$$\sum_{i=1}^{2^{l+1}} q_i = 1 \qquad \text{and} \qquad q_i \geqslant 0, \qquad i = 1, 2,..., 2^{l+1}. \tag{31}$$

After some manipulation, we find that

$$\phi_{l+1} = \phi^0 - \epsilon + 2 \sum_{i=1}^{2^l} \left[(q_i + q_{2^l+i}) \sum_{j=1}^{l} \mu_j(i)\, \epsilon_j u_j \right] + 2 \left(\sum_{i=2^l+1}^{2^{l+1}} q_i \right) \epsilon_{l+1} u_{l+1} . \tag{32}$$

Let

$$\lambda = \sum_{i=2^l+1}^{2^{l+1}} q_i \tag{33}$$

and

$$p_i = q_i + q_{2^l+i}, \qquad i = 1, 2,..., 2^l. \tag{34}$$

Hence, (32) becomes

$$\phi_{l+1} = \phi_l + 2\lambda \epsilon_{l+1} u_{l+1} . \tag{35}$$

With $\lambda = 0$,

$$\phi_{l+1} = \phi_l \in R_c ,$$

by assumption. If $\lambda = 1$,

$$\phi_{l+1} = \phi_l + 2\epsilon_{l+1} u_{l+1} ,$$

which represents a translation of the l-dimensional manifold. Thus, $\phi_{l+1} \in R_c$, by assumption. For $0 < \lambda < 1$, we note that $\phi_{l+1} \in R_c$ if (24)–(26) hold for $j = l + 1$.

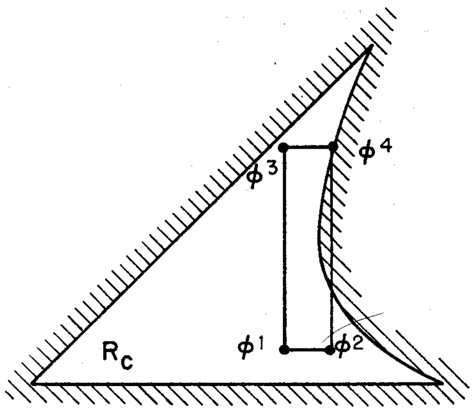

Fig. 6. Possible region R_c.

It is easy to verify that $\phi_1 \in R_c$ and, furthermore, that $\phi_2 \in R_c$ if (24)–(26) hold for $j = 1$ and $j = 2$, respectively. It follows by the foregoing inductive reasoning that $\phi_k = \phi$, as defined by (23), is in R_c under the conditions of the theorem.

The theorem allows both Figs. 4–5, but not Fig. 6.

5. Some Objective Functions

A number of potentially useful and fairly well-behaved objective functions which might be used to represent the cost of a design can be formulated. In practice, of course, a suitable modelling problem would first have to be solved to determine the significant parameters involved partially or totally in the actual cost. Here, we will assume that either absolute or relative tolerances are the main variables and, furthermore, that the total cost $C(\phi^0, \epsilon)$ of the design is just the sum of the cost of the individual components.

It is intuitively reasonable to assume that

$$C(\phi^0, \epsilon) \to c \geqslant 0 \qquad \text{as} \quad \epsilon \to \infty, \tag{36}$$

$$C(\phi^0, \epsilon) \to \infty \qquad \text{for any} \quad \epsilon_i \to 0. \tag{37}$$

Two out of many possible functions which fulfil these requirements are, for $c = 0$,

$$C_a = \sum_{i=1}^{k} (c_i/\epsilon_i), \tag{38}$$

subject to $\epsilon \geqslant 0$ as stated in (8), and

$$C_r = \sum_{i=1}^{k} c_i \log_e(\phi_i{}^0/\epsilon_i), \tag{39}$$

subject to

$$\phi^0 \geqslant \epsilon \geqslant 0. \tag{40}$$

In both cases,

$$c_i \geqslant 0, \qquad i = 1, 2,..., k. \tag{41}$$

6. Examples

It is interesting to consider C_a and C_r for the different regions R_c sketched in Figs. 7–10. We will let $c_1 = c_2 = 1$. Figure 7 depicts a situation where $\check{\phi}^0$ has relatively little variation in going from C_a to C_r. Figure 8 has $\check{\phi}_1{}^0 > \check{\epsilon}_1$ and $\check{\phi}_2{}^0 = \check{\epsilon}_2$; for C_a, $\check{\phi}_2{}^0 > 0$ but, for C_r, $\check{\phi}_2{}^0 = 0$ which (physics permitting) indicates that one parameter may be *removed*. It can be shown (see Fig. 11) that min C_r is given by $\check{\phi}_2{}^0 = 0$, at $\check{\phi}_1{}^0 = 2.5$, $\check{\epsilon}_1 = 1.5$. Figure 9 allows the possibility of removing ϕ_1 if C_r is optimized. The minimum cost is then $\log_e 9$. However, it is easily shown that, to minimize the cost, ϕ_1 should not be removed (see, for example, Fig. 12). Using C_r in Fig. 10 would indicate that $\check{\phi}_1{}^0$ and $\check{\phi}_2{}^0$ may be zero. Using C_a in all the cases of Figs. 7–10, we would find $\check{\phi}^0$ to be an interior point of R_c.

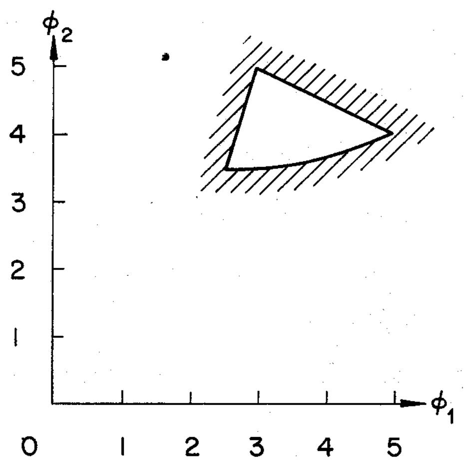

Fig. 7. Example used in the discussion of objective functions.

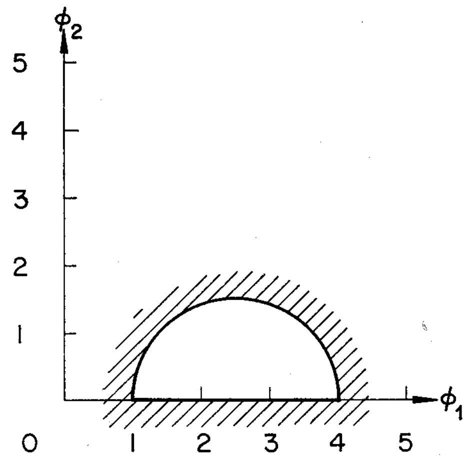

Fig. 8. Example used in the discussion of objective functions.

A number of corresponding observations to those made above can be made if, for the cases sketched in Figs. 7–10, we take (for example) $\phi_1' = 1/\phi_1$ and $\phi_2' = \phi_2$ as parameters.

7. Conclusions

If, as is usual in the design of circuits or systems, the optimal design is obtained by solving an approximation problem, then a fairly

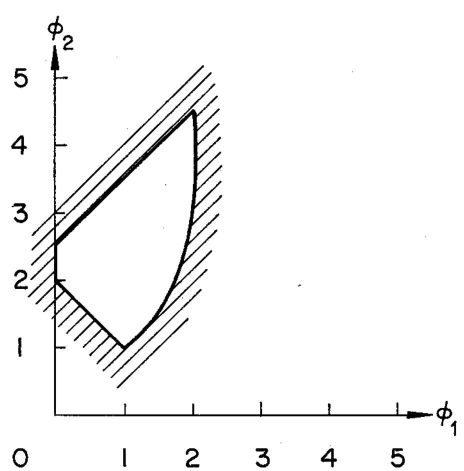

Fig. 9. Example used in the discussion of objective functions.

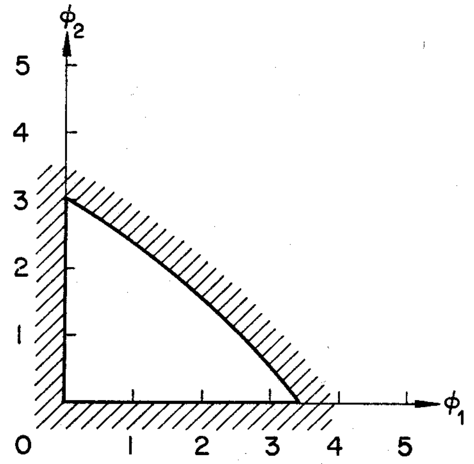

Fig. 10. Example used in the discussion of objective functions.

large number of inequality constraints usually define the acceptable region. For any particular set of reasonable tolerances, one could exploit the likelihood of the worst case (point most likely to violate a given constraint) being predictable by a local linearization or higher-order approximation of the constraints to greatly reduce the computational effort over the computational effort implied by the 2^k vertices of the tolerance region. Further study of these ideas from a nonlinear programming

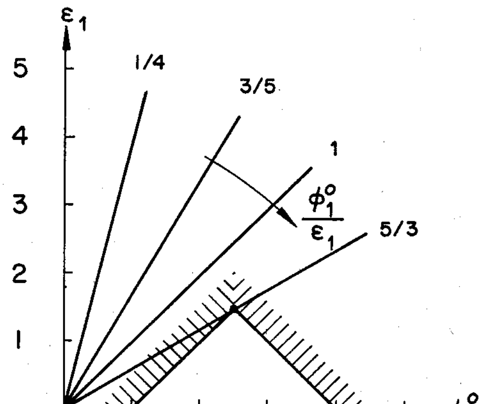

Fig. 11. Example corresponding to Fig. 8 with $\check{\phi}_2{}^0 = \check{\epsilon}_2 = 0$.

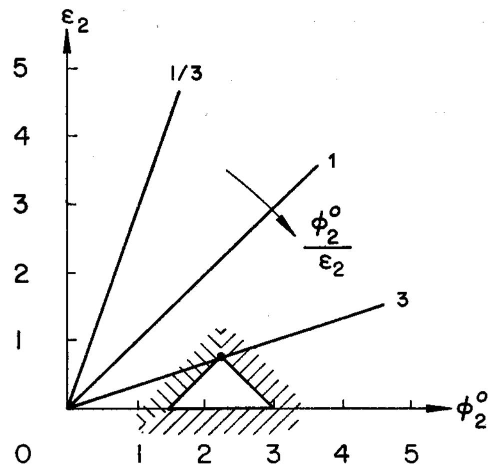

Fig. 12. Example corresponding to Fig. 9 with $\phi_1{}^0 = 1$ and $\epsilon_1 = 0.5$. The best value of C_r is, in this case, $\log_e 6$.

point of view should yield more insight into the possible success or failure of particular tolerance optimization algorithms that might suggest themselves.

References

1. Butler, E. M., *Realistic Design Using Large-Change Sensitivities and Performance Contours*, IEEE Transactions on Circuit Theory, Vol. CT-18, No. 1, 1971.
2. Butler, E. M., *Large Change Sensitivities for Statistical Design*, Bell System Technical Journal, Vol. 50, No. 4, 1971.
3. Karafin, B. J., *The Optimum Assignment of Component Tolerances for Electrical Networks*, Bell System Technical Journal, Vol. 50, No. 4, 1971.
4. Seth, A. K., *Electrical Network Tolerance Optimization*, University of Waterloo, Waterloo, Canada, Ph.D. Thesis, 1972.
5. Pinel, J. F., and Roberts, K. A., *Tolerance Assignment in Linear Networks Using Nonlinear Programming*, IEEE Transactions on Circuit Theory, Vol. CT-19, No. 5, 1972.
6. De Castro, E., Iuculano, G., and Monaco, V. A., *Component Value Spread and Network Function Tolerances: An Optimal Design Procedure*, Alta Frequenza, Vol. 40, No. 11, 1971.

7. MANGASARIAN, O. L., *Nonlinear Programming*, McGraw-Hill Book Company, New York, New York, 1969.
8. ZANGWILL, W. I., *Nonlinear Programming: A Unified Approach*, Prentice-Hall, Englewood Cliffs, New Jersey, 1969.

Printed by the St Catherine Press Ltd., Tempelhof 37, Bruges, Belgium.