DTU Library

A Systematic Procedure for the Generation of Cost-Minimized Designs

Becker, Peter W.; Jarkler, Bjorn

[Link back to DTU Orbit](#)

A Systematic Procedure for the Generation of Cost-Minimized Designs

PETER W. BECKER and BJORN JARKLER

Abstract—We present a procedure for the generation of cost-minimized designs of circuits and systems. Suppose a designer has decided upon the topology of his product. Also suppose he knows the cost and quality of the different grades of the N components required to implement the product. The designer then faces the following problem: How should he proceed to find the combination of grades that will give him the desired manufacturing yield at minimum product cost?

We discuss the problem and suggest a policy by which the designer, with a reasonable computational effort, can find a set of "good" implementations. The suggested policy is applied to an electronic amplifier. The results are quite encouraging.

Reader Aids:

Purpose: Report of software derivation
Special math needed for explanations: Special notation
Special math need for results: Same
Results useful to: Theoretically inclined reliability engineers

I. Introduction

A. The Problem

Suppose a designer has decided upon the topology of his circuit or system (hereafter called product) and knows the cost and quality of the possible choices for each component or subsystem (hereafter called component). Assume, for the moment, that each component is available in three grades: high, medium, and low. For example, a 100-Ω resistor in the design can be realized in hardware by resistors having 1, 5, or 10 percent tolerances, the lower grades presumably having lower costs than the higher grades. When the designer selects a grade for each of the N components, he can do this in 3^N different ways. With each implementation is associated both the cost C and the process reliability R at production time (also called the manufacturing yield) of the resulting product. Each implementation or design (both words will be used to mean a set of N component grades) can be illustrated by a dot, as shown in Fig. 1. Among the huge number of possible implementations, the designer is only interested in those where the yield has been achieved at minimum cost. Such optimal implementations are illustrated by points on the heavy line in the figure. It is among the members of this set of implementations that the designer will select the final design. The question to which we address ourselves in this paper may now be
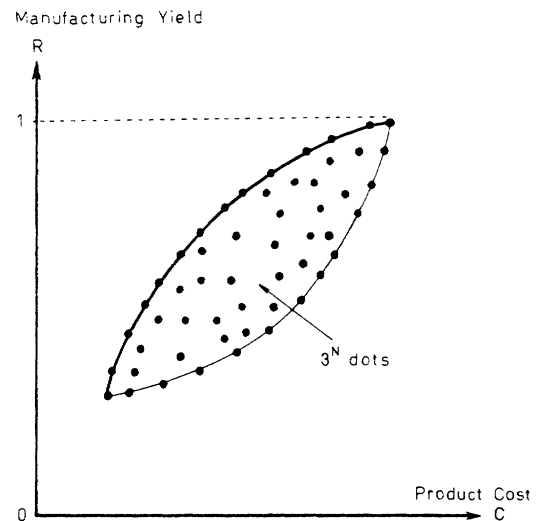
Fig. 1. The interesting implementations. If N components each are available in three different grades, the product can be realized in 3^N ways, as illustrated by the 3^N dots. The designer is only interested in the cost-minimized implementations, illustrated by points on the heavy line.

stated: How can the designer find the optimal implementations without evaluating the C and R of all, or at least an exorbitant number of, implementations?

B. The Organization of the Paper

In Section II we consider the case where all N grades can be changed in infinitesimal steps. Then, under five mild assumptions, we can specify a policy that makes it possible to compute all optimal implementations. In Section III we consider the more realistic case where all N grades can be changed only in finite steps. Again we are able to specify a policy that makes it possible to compute all optimal implementations; however, the computational work becomes unmanageable. In this situation we suggest a suboptimal procedure involving only reasonable computational work. This procedure generates a string of "good" designs.

II. N Grades that Can be Changed in Infinitesimal Steps

A. Five Reasonable Assumptions for the Case $N = 2$

Let us begin by considering the case where the designer varies only the grades of two of the components. The grades are called g_1 and g_2 and are represented in the G plane. We now make five mild assumptions.
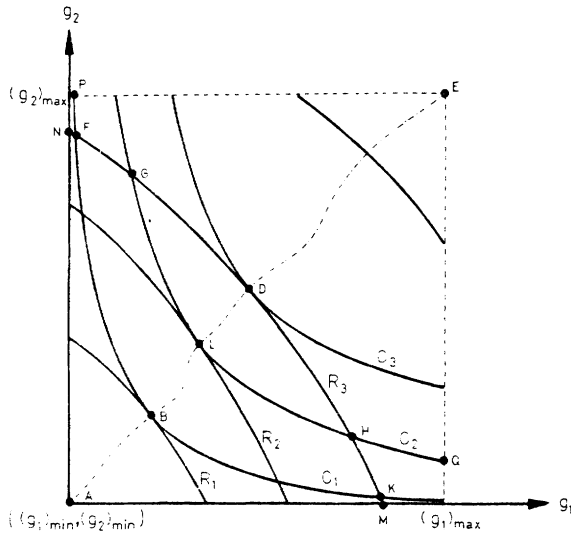
Fig. 2. The locus of cost-minimized implementations. R_1, R_2, and R_3 indicate members of the family of constant R curves. C_1, C_2, and C_3 indicate members of the family of constant C curves. The line $ABLDE$ illustrates the locus of cost-minimized implementations.

Assumption 1: g_i can be measured in a meaningful manner by a scalar, e.g., the grade of the electrical components can be assessed by their tolerance. The scalar has the following two properties: a) the higher the g value, the worse the component, and b) all g values must be greater than zero.

Assumption 2: g_i can be varied in infinitesimal steps between the highest grades $(g_i)_{min}$ and the lowest grades $(g_i)_{max}$. The case where the steps are finite will be treated later.

Assumption 3: C and R are both differentiable functions of g_i in the region of interest.

Assumption 4: If we determine the locus of points in G with the same C value, we obtain a family of convex [1] curves,[1] are illustrated in Fig. 2. The curves are convex since the very best grades cost dramatically more. The radius of curvature at any point is positive; the gradient, in Fig. 2, is in the first quadrant.

Assumption 5: If we determine the locus of points in G with the same R value, we obtain a family of convex curves, as illustrated in Fig. 2. The shape of the curves is determined by the following familiar fact: the effect on process reliability due to the degradation of one component's grade can only be compensated for, to a small degree, by the improvement of some other component's grade. Consequently, the radius of curvature at any point is positive and the gradient is in the first quadrant.

B. The Locus of Cost-Minimized Implementations for N = 2

Consider the region $BFGDHKB$ in Fig. 2. All points in that region have a process reliability of at least R_3 and a product cost of at most C_1. The region is convex because it is an

[1] The reader will recall that convex curves (as well as convex surfaces in three or more dimensions) have the following property: if two points on the curve (or surface) are connected by a chord, then all points on the chord will be located on the same side of the curve (or surface).
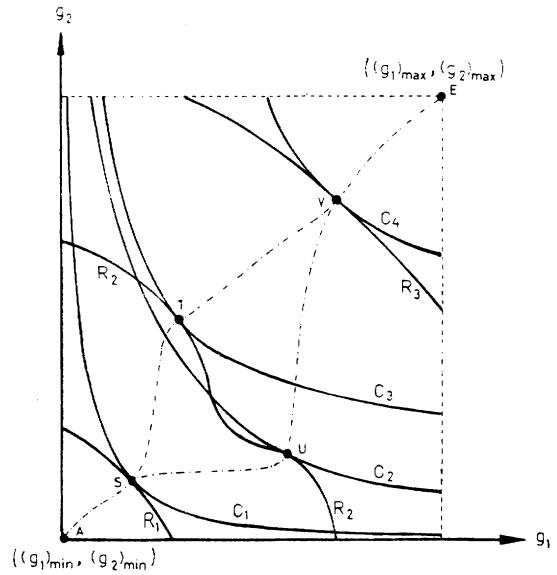

Fig. 3. Bifurcation and buckling. The constant R curve R_2 buckles. This gives rise to bifurcation of the locus of cost-minimized implementations. The implementations SUV are inferior to the implementations STV, e.g., the design illustrated by the point U is inferior to the design illustrated by the point T.

intersection of two convex regions $ANFGDHKMA$ and $EPFBKQE$. If we insist that the process reliability should be at least R_3, and we then try to minimize the product cost, the region will shrink first to the convex region $LGDHL$ and then degenerate to the point D. D represents the least expensive G value with which $R \geqslant R_3$ can be realized, i.e., $R = R_3$ at D. At D the tangents to the C_3 curve and the R_3 curve coincide due to Assumption 3; thus their gradients have the same direction.

Due to the assumption of convexity, the limiting case for the convex intersection was one point D and not two or more separate points; two or more separate points clearly do not constitute a convex region. By repeating the procedure, we can generate a string of optimal implementations A, B, L, D, E, etc. The curve through the points is the locus of cost-minimized implementations that interest the designer. The (C, R) values for the points on the curve are the ones indicated by a heavy line in Fig. 1. The curve does not meander much since the tangent to the curve is always in the first quadrant.

C. Buckling and Bifurcation

In Assumptions 4 and 5 we mentioned that the sets of curves should be convex. The danger of concavities is illustrated by Fig. 3 where a curve buckles and also where the locus of cost-minimized implementations separates into two branches, i.e., bifurcates [2]. Buckling, incidentally, does not necessarily lead to bifurcation. Birfurcation complicates the analysis because it necessitates a study of all alternate paths before the cost-minimzing designs can be determined. The concepts of buckling and bifurcation have lately received increasing attention [2].

D. The Extension to the Case $N > 2$

When $N > 2$, it can be demonstrated [3, Appendix 7], just as in the case $N = 2$, that the locus of cost-minimized

implementations constitutes a path from: 1) the point illustrating the implementation combining maximum reliability and maximum cost to 2) the point illustrating the implementation combining minimum reliability and minimum cost. The proof is based on trivial extensions of the five mild assumptions, and on the geometrical fact that the concept of convexity remains valid for $N > 2$ [1].

E. The Optimal Policy

We can now, under the stated assumptions, formulate an answer to the question that was propounded in Section I-A. The locus of cost-minimized implementations is obtained by: 1) starting with the implementation that combines maximum process reliability and maximum cost, and 2) changing the N grades in such a manner that the cost gradient and the reliability gradient always are proportional; when the gradient vectors have the same direction, we obtain maximum cost savings as we gradually reduce the yield. The locus ends at the implementation that combines minimum yield and minimum cost.

III. N Grades that Can be Changed Only in Finite Steps

A. Suboptimal Designs

When the N grades can be changed in finite steps only, the possible combinations of grades may be illustrated by lattice points in N-dimensional space. Apart from the endpoints, no lattice point will (generally speaking) fall on the locus of cost-minimized implementations. This means that we must, in practice, select our implementations among suboptimal ones. The degree of suboptimality of an implementation with (cost, reliability) = (C_0, R_0) can be described in one of two ways, i.e., one can evaluate: 1) the excess cost $(C_0 - C^*)$ or 2) the reliability deficiency $(R^* - R_0)$ of the implementation where (C_0, R^*) and (C^*, R_0) are the two corresponding cost-minimized implementations. When selecting a suboptimal implementation, we are faced with the problem: Which of two suboptimal implementations (C_1, R_1) and (C_2, R_2) is the better? If, at the same time $C_1 < C_2$ and $R_1 > R_2$, clearly, (C_1, R_1) is superior to (C_2, R_2); but if $C_1 < C_2$ and $R_1 < R_2$, it is an open question whether or not the decrease in yield $(R_2 - R_1)$ was worth the cost saving $(C_2 - C_1)$. This question can only be answered after a study of the particular case.

B. The Suggested Policy

When the N grades can be changed only in finite steps, the optimal policy consists of the following two steps. 1) Discard all implementations that are *inferior* to some other one. This leaves a set of implementations S^* with the following property. If two implementations (C_a, R_a) and (C_b, R_b) both belong to S^*, and $C_a > C_b$, then it is also true that $R_a > R_b$. 2) Among the admissible implementations S^*, select the one that seems best for other reasons, e.g., marketing.

Clearly, the above procedure is unrealistic because it involves computing and comparing a horrendous number of

---

[2] All sums and products in this section are from $i = 1$ to N.

---

implementations, since we cannot obtain S^* in practical cases. We can, however, with a reasonable amount of work, obtain another and smaller set of implementations S that we believe constitute a "good" set of implementations. The suggested procedure is as follows.

Step 1: The first member of S is $\{(g_i)_{\min}\}$ for which C and R both are maximized $(C_{\max}, R_{\max})$.

Step 2: Evaluate (C, R) for the N implementations obtained by reducing the grade g_j, $j = 1, \cdots, N$, by one rating, while maintaining the $(N - 1)$ other grades at maximum level. The result is called $({}_{1}^{j}C, {}_{1}^{j}R)$.

Compute the N values of the fraction ${}_{1}^{j}F$ defined by

$$\,_1^j F \equiv (C_{\max} - \,_1^j C)/(R_{\max} - \,_1^j R). \tag{1}$$

The value j^* that maximizes ${}_{1}^{j}F$ is determined and the corresponding reduction of g_{j^*} by one rating is made permanent; the implementation is called ${}_{1}D$. ${}_{1}D$ is included in S which by now consists of two implementations. The corresponding value of $({}_{1}^{j^*}C, {}_{1}^{j^*}R)$ is renamed $({}_{1}C, {}_{1}R)$.

Step 3: Starting with ${}_{1}D$, repeat Step 2 and obtain implementation ${}_{2}D$ which is included in S. (C, R) for D_2 is named $({}_{2}C, {}_{2}R)$.

Step 4: By repeating the procedure we will arrive at the design $\{(g_i)_{\max}\}$ for which $(C, R) = (C_{\min}, R_{\min})$. This design is also included in S.

If the number of ratings for g_j, is called M_j, then the number of implementations in S is[2]

$$M = 1 + \sum (M_i - 1). \tag{2}$$

How many implementations E must be evaluated before the M members of S have been obtained? A moment's reflection reveals that the largest possible value of E is

$$E_{\max} = 1 + N \sum (M_i - 2) + \sum i. \tag{3}$$

Assuming that $M_1 \leqslant M_2 \leqslant \cdots \leqslant M_N$, it is readily seen that the smallest possible value of E is

$$E_{\min} = 1 + \sum (N + 1 - i)(M_i - 1). \tag{4}$$

In practical cases $E_{\max}$ is much less than the total number of possible implementations ΠM_i.

IV. An Illustrative Example

A. The Three-Transistor Amplifier and its Components

To test the suggested policy, we decided to try it out on a product of reasonable complexity, i.e., the electronic amplifier illustrated in Fig. 4 and described in [4]. It satisfies the five mild assumptions mentioned earlier. The component values listed in Tables I and II were selected to achieve a gain A, $19.5 \leqslant A \leqslant 22.5$ and a 3-dB bandwidth B exceeding 84.5 MHz.

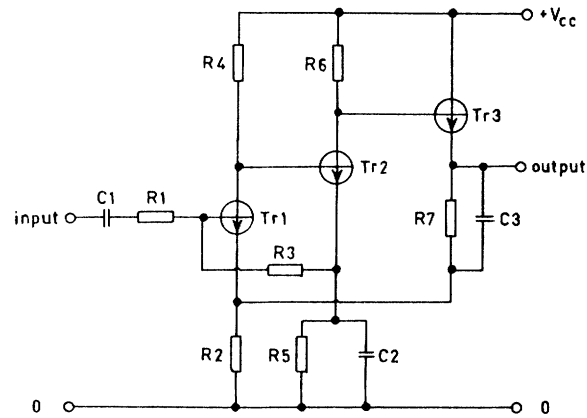A preliminary study of the effect of component grade

Fig. 4. The product. The three-transistor amplifier was used to test the suggested policy.

TABLE I

A Preliminary Study Shows that the Eight Components
Listed Below Should Have a Particular Grade
(One Danish Krone is Approximately 13 Cents)

Component	Grade	Price (Danish kroner)
R_1 = 470 Ω	5 percent	0.29
R_3 = 10 kΩ	5 percent	0.29
R_4 = 22 kΩ	10 percent	0.07
R_5 = 1 kΩ	5 percent	0.29
R_6 = 4.7 kΩ	5 percent	0.29
C_1 = 10 μF	−10 percent, +50 percent	2.00
C_2 = 50 μF	−10 percent, +50 percent	2.00
Transistor 1	2N918	6.30

TABLE II

A Preliminary Study Shows that the Five Components
Listed Below Could Reasonably be Used in Several Grades
(One Danish Krone is Approximately 13 Cents)

Component	Possible Grades	Price (Danish kroner)
R_2 = 22 Ω	1/2 percent tolerance	1.07
	1 percent tolerance	0.85
	5 percent tolerance	0.29
	10 percent tolerance	0.07
R_7 = 470 Ω	1/2 percent tolerance	1.07
	1 percent tolerance	0.85
	5 percent tolerance	0.29
	10 percent tolerance	0.07
C_3 = 15 pF	1.5 percent tolerance	0.50
	5 percent tolerance	0.40
Transistor 2	2N918	6.30
	2N918 selected	8.30
Transistor 3	2N918	6.30
	2N2369	3.90

changes on the output specifications showed that some of the components obviously should have a particular grade to obtain a cost-minimized product design; eight such components are listed in Table I. For some of the other components, several grades seemed to be reasonable choices; five such components are listed in Table II. We felt that the range endpoints of true values was the most reliable information we had on the resistors, so we assumed the resistors to have the rectangular (also called flat or uniform) distributions [3, Sec. 6.3]. With each transistor are associated values of r_{bb}', $r_b'e$, r_{ce}, g_m, $C_b'e$, and $C_b'c$ that are statistically dependent. Fortunately, corresponding transistor parameter values for 100 2N918 transistors, as well as for 100 2N2369 transistors, have been tabulated in the literature [5]; the tabulations cover nine operating points per transistor. By "2N918 selected" in Table II we mean that 2N918 transistors have had the $C_b'c$ value measured and only the best half is used; the selected transistor had $C_b'c$ less than 0.6151 pF.

B. Exhaustive Search Versus the Suggested Policy

An inspection of Table II shows that the total number of ways of implementing the product is $4 \cdot 4 \cdot 2 \cdot 2 \cdot 2 = 128$. The number is moderate and we decided to perform an exhaustive search for good implementations. For each possible implementation we determined two quantities: the total cost C for the components from Table II and the joint probability R of the product's simultaneously meeting the two output specifications, $19.5 \leqslant A \leqslant 22.5$ and $B > 84.5$ MHz. Due to grade variations, C is the variable part of the product cost, so we are only concerned with changes in C values. The (C, R) values for the 128 implementation methods are illustrated as points in Fig. 5. Twenty-four of the points coincide with some of the remaining 104 points. Fig. 5 clearly shows the danger of getting an inferior implementation. The excess cost $(C - C^*)$ can easily be a dime or more for an implementation that should have a specified yield and that was selected carelessly. The members of the S^* and S are illustrated in Fig. 5.

When we compare the encircled points in Fig. 5 with the remaining points, it is evident that the suggested policy results in a small but a "good" set S of M implementations. What makes the set S "good" is that any member of S^* equals, or is "almost inferior" to, some member of S, the only noticeable exception being the implementation $(18.36, 86.0)$. The implementation has the following grades:

$(R_2, R_7, C_3,$ Transistor 2, Transistor 3)

$= (0.5$ percent, 5 percent, 1.5 percent, 2N918, 2N2369). (5)

Due to the fact that S^* contains substantially more implementations than S, it seems unavoidable that S^* contain some implementations to which no member of S is equal or "almost superior."

V. Conclusions

The problem of selecting an implementation that minimizes cost for a desired yield has been studied. A policy has been suggested for finding a "good" set of implementations with a
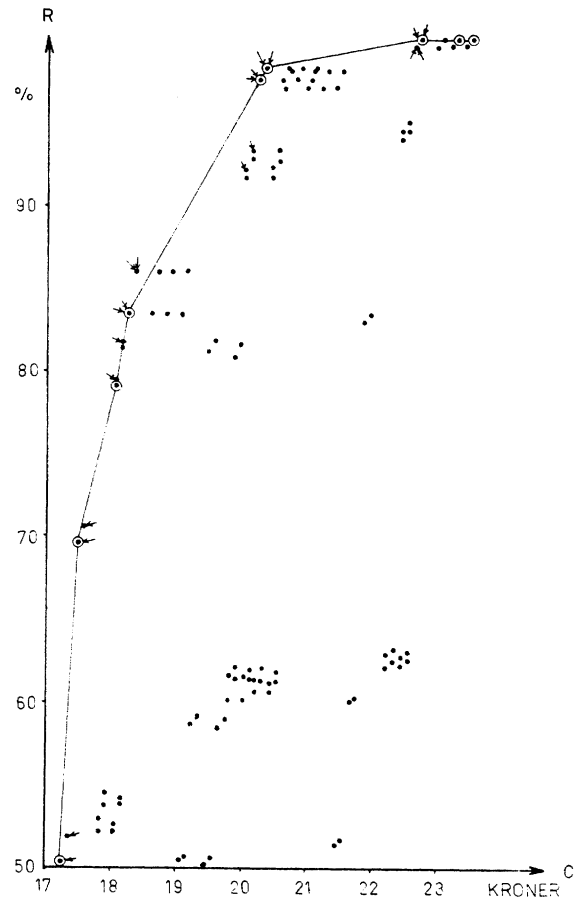

Fig. 5. A "good" set of implementations. The points illustrate 126 of the 128 possible implementations from the example; the points (17.14, 44.5) and (17.04, 43.1) have been omitted. By following the suggested policy, we find the ten members of S, one of which is (17.04, 43.1); the implementations are illustrated by encircled points that for illustrative purposes are connected by straight lines. The 20 arrows point to 20 members of S^*, the two remaining members of S^* being the omitted points. Six of the points each illustrate two members of S^*. (One Danish krone equals approximately 13 cents.)

reasonable amount of computational work. The designer can either settle for one of the "good" implementations or use the "good" set as a starting point in a search for a more suitable implementation.

References

[1] L.A. Lyusternik, *Convex Figures and Polyhedra.* New York: Dover, 1963, article 7.7.

[2] J.B. Keller and S. Antman, *Bifurcation Theory and Nonlinear Eigenvalue Problems.* New York: Courant Inst. Math. Sci., New York Univ., 1968.

[3] P.W. Becker and F. Jensen, *Yield and Reliability Optimization of Electronic Systems and Circuits,* to be published.

[4] "Applications of the E-line epoxy encapsulated transistor," Electron. Dep.., Ferranti, Ltd., United Kingdom, Appl. Notes, pp. 25-27, Nove. 1967.

[5] *Racal Electron. Des. and Anal. by Comput., Measurements Manual,* vol. 1. United Kingdom: Racal Research Ltd., 1968.