SIMULATION OPTIMIZATION SYSTEMS
Research Laboratory

CIRCUIT OPTIMIZATION:
THE STATE OF THE ART

J.W. Bandler and S.H. Chen

SOS-87-2-R

April 1987

McMASTER UNIVERSITY
Hamilton, Canada L8S 4L7
Department of Electrical and Computer Engineering

CIRCUIT OPTIMIZATION: THE STATE OF THE ART

J.W. Bandler and S.H. Chen

SOS-87-2-R

April 1987

CIRCUIT OPTIMIZATION: THE STATE OF THE ART

J.W. Bandler, Fellow, IEEE, and S.H. Chen, Student Member, IEEE

Invited Paper

Abstract This paper reviews the current state of the art in circuit optimization, emphasizing techniques suitable for modern microwave CAD. It is directed at the solution of realistic design and modeling problems, addressing such concepts as physical tolerances and model uncertainties. A unified hierarchical treatment of circuit models forms the basis of the presentation. It exposes tolerance phenomena at different parameter/response levels. The concepts of design centering, tolerance assignment and postproduction tuning in relation to yield enchancement and cost reduction suitable for integrated circuits are discussed. Suitable techniques for optimization oriented worst-case and statistical design are reviewed. A generalized ℓ_p centering algorithm is proposed and discussed. Multi-circuit optimization directed both at CAD and robust device modeling is formalized. Tuning is addressed in some detail, both at the design stage and for production alignment. Practical examples covering a wide range of microwave circuit problems illustrate the relevant concepts. State of the art gradient-based nonlinear optimization methods are reviewed with emphasis given to recent, but well-tested, advances in minimax, ℓ_1 and ℓ_2 optimization. Useful formulas for sensitivity calculations and gradient approximations are also presented.

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under Grant A7239.

The authors are with the Simulation Optimization Systems Research Laboratory and the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada L8S 4L7.

J.W. Bandler is also with Optimization Systems Associates Inc., 163 Watson's Lane, Dundas, Ontario, Canada L9H 6L1.

I. INTRODUCTION

Computer-aided circuit optimization is certainly one of the most active areas of interest. Its advances continue, hence the subject deserves regular review from time to time. The classic paper by Temes and Calahan in 1967 [116] was one of the earliest to formally advocate the use of iterative optimization in circuit design. Techniques that were popular at the time, such as one-dimensional (single-parameter) search, the Fletcher-Powell procedure and the Remez method for Chebyshev approximation, were described in detail and well-illustrated by circuit examples. Pioneering papers by Lasdon, Suchman and Waren [83,84,121] demonstrated optimal design of linear arrays and filters using the penalty function approach. Two papers in 1969 by Director and Rohrer [56,57] originated the adjoint network approach to sensitivity calculations, greatly facilitating the use of powerful gradient-based optimization methods. In the same period, the work by Bandler [6,7] systematically treated the formulation of error functions, the least pth objective, nonlinear constraints, optimization methods and circuit sensitivity analysis.

Since then, advances have been made in several major directions. The development of large-scale network simulation and optimization techniques has been motivated by the requirements of the VLSI era. Approaches to realistic circuit design where design parameter tolerances and yield are taken into account have been pioneered by Elias [60] and Karafin [79] and furthered by many authors over the ensuing years. Optimization methods have evolved from simple, low-dimension-oriented algorithms into sophisticated and powerful ones. Highly effective and efficient solutions have been found for a large number of specialized applications. The surveys by Calahan [43], Charalambous [44], Bandler and Rizk [31], Hachtel and Sangiovanni-Vincentelli [73], and Brayton et al. [38] are especially relevant to circuit designers.

In the present paper, we concentrate on aspects that are relevant to and necessary for the continuing move to optimization of increasingly more complex microwave circuits, in particular to MMIC circuit modeling and design. Consequently, we emphasize optimization

oriented approaches to deal more explicitly with process imprecision, manufacturing tolerances, model uncertainties, measurement errors, and so on. Such realistic considerations arise from design problems in which a large volume of production is envisaged, e.g., integrated circuits. They also arise from modeling problems in which consistent and reliable results are expected despite measurement errors, structural limitations such as physically inaccessible nodes, and model approximations and simplifications. The effort to formulate and solve these problems represents one of the driving forces of theoretical study in the mathematics of circuit CAD. Another important impetus is provided by progress in computer hardware, resulting in drastic reduction in the cost of mass computation. Finally, the continuing development of gradient-based optimization techniques has provided us with powerful tools.

In this context, we review the following concepts: realistic representations of a circuit design and modeling problem, nominal (single) circuit optimization, statistical circuit design and multi-circuit modeling, recent gradient-based optimization methods, as well as the calculation and approximation of gradients for the optimizers.

Nominal design and modeling are the conventional approaches used by microwave engineers. Here, we seek a single point in the space of variables selected for optimization which best meets a given set of performance specifications (in design) or best matches a given set of response measurements (in modeling). A suitable scalar measure of the deviation between responses and specifications which forms the objective function to be minimized is the ubiquitous least squares measure (see, for example, Morrison [93]), the more esoteric generalized ℓ_p objective (Charalambous [46]) or the minimax objective (Madsen et al. [90]). We observe here that the performance driven (single circuit) least squares approach that circuit design engineers have traditionally chosen has proved unsuccessful both in addressing design yield as well as in serious device modeling.

Recognition that an actual realization of a nominal design is subject to fluctuation or deviation led, in the past, to the so-called sensitivity minimization approach (see, for example,

Schoeffler [108] and Laker et al. [81]. Employed by filter designers, the approach involves measures of performance sensitivity, typically first-order, and including it in the objective function.

In reality, uncertainties which deteriorate performance may be due to physical (manufacturing, operating) tolerances as well as parasitic effects such as electromagnetic coupling between elements, dissipation and dispersion (Bandler [8], Tromp [120]). In the design of substantially untunable circuits these phenomena lead to two important classes of problems: worst-case design and statistical design. The main objective is the reduction of cost or the maximization of production yield.

Worst-cast design (Bandler et al. [28,29]), in general, requires that all units meet the design specifications under all circumstances (i.e., a 100% yield), with or without tuning, depending on what is practical. In statistical design [1,31,36,55,111,112,114,115] it is recognized that a yield of less than 100% is likely and therefore, with respect to an assumed probability distribution function, yield is estimated and enhanced by optimization. Typically, we either attempt to center the design with fixed assumed tolerances or we attempt to optimally assign tolerances and/or design tunable elements to reduce production cost.

What distinguishes all these problems from nominal designs or sensitivity minimization is the fact that a single design point is no longer of interest: a (tolerance) region of multiple possible outcomes is to be optimally located with respect to the acceptable (feasible, constraint) region.

Modeling, often unjustifiably treated as if it were a special case of design, is particularly affected by uncertainties and errors at many levels. Unavoidable measurement errors, limited accessibility to measurement points, approximate equivalent circuits, etc., result in nonunique and frequently inconsistent solutions. To overcome these frustrations, we advocate a properly constituted multi-circuit approach (Bandler et al. [16]).

Our presentation is organized into nine sections.

In Section II, in relation to a physical engineering system of interest, a typical hierarchy of simulation models and corresponding response and performance functions are introduced. Error functions arising from given specifications and a vector of optimization variables are defined. Performance measures such as ℓ_p objective functions (ℓ_p norms and generalized ℓ_p functions) are introduced and their properties discussed.

We devote to Section III a brief review of the relatively well-known and successful approach of nominal circuit design optimization. Illustrative examples are also given.

In Section IV, uncertainties that exist in the physical system and at different levels of the model hierarchy are discussed and illustrated by a practical example. Different cases of multi-circuit design, namely centering, tolerancing (optimal tolerance assignment) and tuning at the design stage, are identified. A multi-circuit modeling approach and several possible applications are described.

Some important and representative techniques in worst-case and statistical design are reviewed in Section V. These include the nonlinear programming approach to worst-case design (Bandler et al. [29], Polak [99]), simplicial (Director and Hachtel [55]) and multi-dimensional (Bandler and Abdel-Malek [9]) approximations of the acceptable region, the gravity method (Soin and Spence [112]) and the parametric sampling method (Singhal and Pinel [111]). A generalized ℓ_p centering algorithm is proposed as a natural extension to ℓ_p nominal design. It provides a unified formulation of yield enchancement for both the worst-case and the case where yield is less than 100%.

Illustrations of statistical design are given in Section VI.

The studies in the last two decades on the theoretical and algorithmic aspects of optimization techniques have produced a great deal of results. Especially, gradient-based optimization methods have gained increasing popularity in recent years for their effectiveness and efficiency. The essence of the important Gauss-Newton, quasi-Newton and conjugate gradient families, which encompass the majority of gradient-based methods, is reviewed in Section VII. Emphasis is given to the use of a trust region and the solution of

optimality equations by a quasi-Newton iteration (Madsen [88], Moré [92], Dennis and Moré [54]). Special attention is also given to minimax and ℓ_1 optimization when the objective function is not differentiable (Hald and Madsen [75,76]).

Gradient calculations for linear circuits are introduced in Section VIII. Sensitivity formulas for networks with a nodal description (Director and Rohrer [56], Branin [37]), unterminated and terminated two-ports, as well as some commonly used frequency responses are presented. Obstacles in practical implementation relating to large-scale networks are addressed. A brief treatment of the subject of gradient approximation (Bandler et al. [19,20,21]) is also given.

Section IX contains our conclusions to this paper and some comments on future research directions.

II. VARIABLES AND FUNCTIONS

In this section, we review some basic concepts of practical circuit optimization. In particular, we identify a physical system and its simulation models. We discuss a typical hierarchy of models and the associated designable parameters and response functions. We also define specifications, error functions, optimization variables and objective functions.

The Physical System

The physical engineering system under consideration can be a network, a device, a process, and so on, which has both a fixed structure and given element types. We manipulate the system through some adjustable parameters contained in the column vector $\boldsymbol{\Phi}^{M}$. The superscript M identifies concepts related to the physical system. Geometrical dimensions such as the width of a strip and the length of a waveguide section are examples of adjustable parameters.

In the production of integrated circuits, $\boldsymbol{\Phi}^{M}$ may include some fundamental variables which control, say, a doping or photomasking process and, consequently, determine the

geometrical and electrical parameters of a chip. External controls, such as the biasing voltages applied to an active device, are also possible candidates for Φ^M.

The performance and characteristics of the system are described in terms of some measurable quantities. The usual frequency and transient responses are typical examples. These measured responses, or simply measurements, are denoted by $F^M(\Phi^M)$.

The Simulation Models

In circuit optimization, some suitable models are used to simulate the physical system. Actually, models can be usefully defined at many levels. Tromp [119,120] has considered an arbitrary number of levels (also see Bandler et al. [24]). (It is our opinion that a substantial reason for the failures in the practical implementation of CAD and device modeling is the casual attitude taken to the identification and implementation of parameters and responses.) Here, for simplicity, we consider a hierarchy of models consisting of four typical levels as

$$\begin{aligned} F^H &= F^H(F^L), \\ F^L &= F^L(\Phi^H), \\ \Phi^H &= \Phi^H(\Phi^L). \end{aligned} \tag{1}$$

Φ^L is a set of low-level model parameters. It is supposed to represent, as closely as possible, the adjustable parameters in the actual system, i.e., Φ^M. Φ^H defines a higher-level model, typically an equivalent circuit, with respect to a fixed topology. Usually, we use an equivalent circuit for the convenience of its analysis. The relationship between Φ^L and Φ^H is either derived from theory or given by a set of empirical formulas.

Next on the hierarchy we define the model responses at two possible levels. The low-level external representation, denoted by F^L, can be the frequency-dependent complex scattering parameters, unterminated y-parameters, transfer function coefficients, etc. Although these quantities may or may not be directly measurable, they are very often used to

represent a subsystem. The high-level responses $\mathbf{F}^H$ directly correspond to the actual measured responses, namely $\mathbf{F}^M$, which may be, for example, frequency responses such as return loss, insertion loss and group delay of a suitably terminated circuit.

A realistic example of a one-section transformer on stripline was originally considered by Bandler et al. [30]. The circuits and parameters, physical as well as model, are shown in Fig. 1. The physical parameters $\boldsymbol{\phi}^M$ (and the low-level model $\boldsymbol{\phi}^L$) include strip widths, section lengths, dielectric constants, strip and substrate thicknesses. The equivalent circuit has six parameters, considered as $\boldsymbol{\phi}^H$, including the effective linewidths, junction parasitic inductances and effective section length. The scattering matrix of the circuit with respect to idealized (matched) terminations is a candidate for a low-level external representation ($\mathbf{F}^L$). The reflection coefficient by taking into account the actual complex terminations could be a high-level response of interest ($\mathbf{F}^H$).

For a particular case, we may choose a certain section of this hierarchy to form a design problem. We can choose either $\boldsymbol{\phi}^L$ or $\boldsymbol{\phi}^H$ as the designable parameters. Either $\mathbf{F}^L$ or $\mathbf{F}^H$ or a suitable combination of the both may be selected as the response functions. Bearing this in mind, we simplify the notation by using $\boldsymbol{\phi}$ for the designable parameters and $\mathbf{F}$ for the response functions.

Specifications and Error Functions

The following discussion on specifications and error functions is based on presentations by Bandler [7], and Bandler and Rizk [31], where more exhaustive illustrations can be found.

We express the desirable performance of the system by a set of specifications which are usually functions of certain independent variable(s) such as frequency, time, temperature, etc. In practice, we have to consider a discrete set of samples of the independent variable(s) such that satisfying the specifications at these points implies satisfying them almost everywhere. Also, we may consider simultaneously more than one kind of response. Thus, without

loss of generality, we denote a set of sampled specifications and the corresponding set of calculated response functions by, respectively,

$$\begin{aligned} &S_j\,, && j = 1, 2, \ldots, m\,, \\ &F_j(\Phi)\,, && j = 1, 2, \ldots, m\,. \end{aligned} \tag{2}$$

Error functions arise from the difference between the given specifications and the calculated responses. In order to formulate the error functions properly, we may wish to distinguish between having upper and lower specifications (windows) and having single specifications, as illustrated in Figs. 2-a and 3-a. Sometimes the one-sidedness of upper and lower specifications is quite obvious such as in the case of designing a bandpass filter. On other occasions the distinction is more subtle, since a single specification may as well be interpreted as a window having zero width.

In the case of having single specifications, we define the error functions by

$$e_j(\Phi) = w_j |F_j(\Phi) - S_j|\,, \qquad j = 1, 2, \ldots, m\,, \tag{3}$$

where w_j is a nonnegative weighting factor.

We may also have an upper specification S_{uj} and a lower specification $S_{\ell j}$. In this case we define the error functions as

$$\begin{aligned} e_{uj}(\Phi) &= w_{uj}(F_j(\Phi) - S_{uj})\,, && j \in J_u\,, \\ e_{\ell j}(\Phi) &= w_{\ell j}(F_j(\Phi) - S_{\ell j})\,, && j \in J_\ell\,. \end{aligned} \tag{4}$$

where w_{uj} and $w_{\ell j}$ are nonnegative weighting factors. The index sets as defined by

$$\begin{aligned} J_u &= \{j_1, j_2, \ldots, j_k\}\,, \\ J_\ell &= \{j_{k+1}, j_{k+2}, \ldots, j_m\}\,. \end{aligned} \tag{5}$$

are not necessarily disjoint (i.e., we may have simultaneous specifications). In order to have a set of uniformly indexed error functions, we let

$$\begin{aligned} e_i &= e_{uj}(\Phi)\,, && j = j_i\,, \quad i = 1, 2, \ldots, k\,, \\ e_i &= -e_{\ell j}(\Phi), && j = j_i\,, \quad i = k+1, k+2, \ldots, m\,. \end{aligned} \tag{6}$$

The responses corresponding to the single specifications can be real or complex, whereas upper and lower specifications are applicable to real responses only. Notice that, in

either case, the error functions are real. Clearly, a positive (nonpositive) error function indicates a violation (satisfaction) of the corresponding specification. Figures 2-b and 3-b depict the concept of error functions.

Optimization Variables and Objective Functions

Mathematically, we abstract a circuit optimization problem by the following statement

$$\underset{\mathbf{x}}{\text{minimize}}\ U(\mathbf{x}) \tag{7}$$

where $\mathbf{x}$ is a set of optimization variables and $U(\mathbf{x})$ a scalar objective function.

Optimization variables and model parameters are two separate concepts. As will be elaborated on later in this paper, $\mathbf{x}$ may contain a subset of ϕ which may have been normalized or transformed, it may include some statistical variables of interest, several parameters in ϕ may be tied to one variable in $\mathbf{x}$, and so on.

Typically, the objective function $U(\mathbf{x})$ is closely related to an ℓ_p norm or a generalized ℓ_p function of $\mathbf{e}(\phi)$. We shall review the definitions of such ℓ_p functions and discuss their appropriate use in different contexts.

The ℓ_p Norms

The ℓ_p norm (Temes and Zai [117]) of $\mathbf{e}$ is defined as

$$\| \mathbf{e} \|_p = \left[\sum_{j=1}^{m} |e_j|^p \right]^{1/p} . \tag{8}$$

It provides a scalar measure of the deviations of the model responses from the specifications. Least-squares (ℓ_2) is perhaps the most well-known and widely used norm (Morrison [93]), which is

$$\| \mathbf{e} \|_2 = \left[\sum_{j=1}^{m} |e_j|^2 \right]^{1/2} . \tag{9}$$

The ℓ_2 objective function is differentiable and its gradient can be easily obtained from the partial derivatives of e. Partly due to this property, a large variety of ℓ_2 optimization techniques have been developed and popularly implemented. For example, the commercial CAD packages TOUCHSTONE [118] and SUPER-COMPACT [113] have hitherto provided designers soley the least-squares objective.

The parameter p has an important implication. By choosing a large (small) value for p, we in effect place more emphasis on those error functions (e_j's) that have larger (smaller) values. By letting $p = \infty$ we have the minimax norm

$$\| e \|_\infty = \max_j |e_j| \tag{10}$$

which directs all the attention to the worst case and the other errors are in effect ignored. Minimax optimization is extensively employed in circuit design where we wish to satisfy the specifications in an optimal equal-ripple manner [3,17,18,26,45,47,75,78,90,95].

On the other hand, the use of the ℓ_1 norm,as defined by

$$\| e \|_1 = \sum_{j=1}^{m} |e_j|, \tag{11}$$

implies attaching more importance to the error functions that are closer to zero. This property has led to the application of ℓ_1 to data-fitting in the presence of gross errors [27,35,76,96] and, more recently, to fault location [10,11,33] and robust device modelling [16].

Notice that neither $\|e\|_\infty$ nor $\|e\|_1$ is differentiable in the ordinary sense. Therefore, their minimization requires algorithms that are much more sophisticated than those for the ℓ_2 optimization.

The One-sided and Generalized ℓ_p Functions

By using an ℓ_p norm, we try to minimize the errors towards a zero value. In cases where we have upper and lower specifications, a negative value of e_j simply indicates that the specification is exceeded at that point which is, in a sense, better than having $e_j = 0$. This fact leads to the one-sided ℓ_p function defined by

$$H_p^+(e) = \left[\sum_{j \in J} |e_j|^p \right]^{1/p}, \tag{12}$$

where $J = \{j \mid e_j \geq 0\}$. Actually, if we define $e_j^+ = \max\{e_j, 0\}$, then $H_p^+(e) = \|e^+\|_p$.

Bandler and Charalambous [12,46] have proposed the use of a generalized ℓ_p function defined by

$$H_p(e) = \begin{cases} H_p^+(e) & \text{if the set } J \text{ is not empty} \\ H_p^-(e) & \text{otherwise} \end{cases} \tag{13}$$

where

$$H_p^-(e) = -\left[\sum_{j=1}^{m} (-e_j)^{-p} \right]^{-1/p}. \tag{14}$$

In other words, when at least one of the e_j is nonnegative we use H_p^+, and H_p^- is defined if all the error functions have become negative.

Compared to (12), the generalized ℓ_p function has an advantage in the fact that it is meaningfully defined for the case where all the e_j are negative. This permits its minimization to proceed even after all the specifications have been met, so that the specifications may be further exceeded. The current versions of TOUCHSTONE [118] and SUPER-COMPACT [113] provide only the one-sided ℓ_2 objective function, thus the optimization stops once the specifications are met.

A classical example is the design of Chebyshev type bandpass filters, where we have to minimize the generalized minimax function

$$H_\infty(e) = \max_j \{e_j\}. \tag{15}$$

Very recently, Bandler et al. [27] have reported the use of the one-sided ℓ_1 function in network design with an aim of providing a good starting point for subsequent minimax optimization.

The Acceptable Region

We use H(e) as a generic notation for $\|e\|_p$, $H_p^+(e)$ and $H_p(e)$. The sign of $H(e(\phi))$ indicates whether or not all the specifications are satisfied by ϕ. An acceptable region is defined as

$$R_a = \{\phi \mid H(e(\phi)) \leq 0\}, \tag{16}$$

Figures 2-c, 2-d, 3-c and 3-d depict the ℓ_p functions and the acceptable regions.

III. NOMINAL CIRCUIT OPTIMIZATION

Introduction

In a nominal design, without considering tolerances (i.e., assuming that modeling and manufacturing can be done with absolute accuracy), we seek a single set of parameters, called a nominal point and denoted by ϕ^0, which satisfies the specifications. Furthermore, if we consider the functional relationship of $\phi^H = \phi^H(\phi^L)$ to be precise, then it does not really matter at which level the design is conceived. In fact, traditionally it is often oriented to an equivalent circuit.

A classical case is network synthesis where $\phi^{H,0}$ is obtained through the use of an equivalent circuit and/or a transfer function. A low-level model $\phi^{L,0}$ is then calculated from $\phi^{H,0}$, typically with the help of an empirical formula (e.g., the number of turns of a coil is calculated for a given inductance). Finally, we try to realize $\phi^{L,0}$ by its physical counterpart $\phi^{M,0}$.

With the tool of mathematical optimization, the nominal point ϕ^0 (at a chosen level) is obtained through the minimization of $U(\mathbf{x})$, where the objective function is typically defined as an ℓ_p function $H(\mathbf{e})$. The vector $\mathbf{x}$ contains all the elements of or a subset of the elements of ϕ^0. It is a common practice to have some of the variables normalized. It is also common to have several model parameters tied to a single variable. This is true, e.g., for symmetrical circuit structures but, most importantly, it is a fact of life in integrated circuits. Indeed, such

dependencies should be taken into account both in design and modeling to reduce the dimensionality.

Manifold Multiplexer Example

As illustrations, we consider nominal design optimization of contiguous band microwave multiplexers consisting of multi-coupled cavity filters distributed along a waveguide manifold. This has been a problem of significant interest (Atia [4], Chen et al. [48,49]). We devote our attention to description of the circuits and set-up of the design problems, while leaving the mathematics of the optimization techniques to Section VII.

A typical multiplexer structure is shown in Fig. 4. Recently, a general approach to simulation and sensitivity analysis of multiplexers has been presented in [23]. A major task in multiplexer optimization is to design the channel filters and to determine the location of these filters along the waveguide manifold. The responses of interest for a typical problem are common-port return loss and insertion loss between common port and channel output ports.

The physical parameters for a channel filter include the geometrical dimensions of the cross slots through which the cavities are coupled and the penetrations of the coupling screws by which different modes in the same cavity are coupled. The cavity resonant frequencies may also be adjusted using tuning screws. The narrowband equivalent circuit introduced by Atia and Williams [5] can be used where we consider, as high-level model parameters, a coupling matrix as well as input and output transformer ratios. Exact sensitivity formulas for such filters have been given by Bandler et al. [15].

The interface between the filter and manifold waveguides can be a T-junction for which an empirical model due to Chen et al. [49] may be used. The location of channel filters along the manifold can be described by spacings. They prove to be critical parameters in design.

The first example is a 12 GHz, 12 channel multiplexer recently described by Bandler, Kellermann and Madsen [27]. It is a contiguous band multiplexer having a channel frequency separation of 40 MHz and a usable bandwidth of 39 MHz. The center frequency of channel No. 1 is 12180 MHz. Twelve 6th order filters are used which have designable couplings indicated in the following matrix

$$\mathbf{M} = \begin{bmatrix} M_{11} & M_{12} & 0 & 0 & 0 & 0 \\ M_{12} & M_{22} & M_{23} & 0 & 0 & 0 \\ 0 & M_{23} & M_{33} & M_{34} & 0 & M_{36} \\ 0 & 0 & M_{34} & M_{44} & M_{45} & 0 \\ 0 & 0 & 0 & M_{45} & M_{55} & M_{56} \\ 0 & 0 & M_{36} & 0 & M_{56} & M_{66} \end{bmatrix}$$

The diagonal elements M_{ii} represent deviations from the synchroneously tuned resonance. Notice that **M** is symmetrical such that M_{ij} and M_{ji} depend on the same variable. Elements of **M** may be considered as Φ^H. A lower specification of 20 dB on the common port return loss is imposed over the passband of all 12 channels. As shown in Fig. 5, the specification is seriously violated at the starting point.

The design process is started with a one-sided ℓ_1 optimization using the algorithm in [27]. We use ℓ_1 to deemphasize the worst violations of the specification and concentrate on the smaller errors. At this stage only the waveguide spacings are optimized since they are considered as the dominant variables of the problem, based on an initial sensitivity analysis. The spacing between the kth and $(k-1)$th channels is initially set to half the wavelength corresponding to the kth center frequency. The result of the one-sided ℓ_1 optimization is shown in Fig. 6. Serious violations of the specification occur in the frequency range corresponding to channels 1-2 and 8-12. This motivates us to release additional variables associated with these channels including filter couplings and transformer ratios. From that

point minimax optimization [26] is employed involving 60 variables. The final optimized return loss is shown in Fig. 7.

As a second example, we consider expanding the previous 12 channel design into a 16 channel multiplexer. It involves 240 nonlinear designable parameters. Instead of making a blind attempt to optimize all the variables simultaneously, we utilize the new decomposition approach to large scale problems described in [18,34]. By adding one channel at a time, we are able to define a localized problem involving a relatively small number of variables and functions. For example, when the 13th channel is added, we optimize only variables in channels 12 and 13 with specifications imposed on responses in channels 11, 12 and 13. Although this may be justified intuitively, it is actually soundly based on sensitivity analyses. By repeating such a decomposition process four times, in which channels 13–16 are added and optimized successively, an optimal design is reached, as shown in Fig. 8.

Optimization of non-contiguous band multiplexers has also been considered in [26].

Extensions of Nominal Design to Modeling

Traditionally, the approach of nominal design has been extended to solving modeling problems. A set of measurements made on the physical system serves as single specifications. Error functions are created from the differences between the calculated responses $\mathbf{F}(\boldsymbol{\phi}^0)$ and the measured responses $\mathbf{F}^M$. By minimizing an ℓ_p norm of the error functions, we attempt to identify a set of model parameters $\boldsymbol{\phi}^0$ such that $\mathbf{F}(\boldsymbol{\phi}^0)$ best matches $\mathbf{F}^M$. This is known as data fitting or parameter identification.

Such a casual treatment of modeling as if it were a special case of design is often unjustifiable, due to the lack of consideration to the uniqueness of the solution. In design, one satisfactory nominal point, possibly out of many feasible solutions, may suffice. In modeling, however, the uniqueness of the solution is almost always essential to the problem. Affected by uncertainties at many levels, unavoidable measurement errors and limited accessibility to measurement points, the model obtained by a nominal optimization is often nonunique and

unreliable. To overcome these frustrations, a recent multi-circuit approach will be described in Section IV.

IV. A MULTI-CIRCUIT APPROACH

The approach of nominal circuit optimization, which we have described in Section III, focuses attention on a certain kind of idealized situation. In reality, unfortunately, there are many uncertainties to be accounted for. For the physical system, without going into too many details, consider

$$\begin{aligned} \mathbf{F}^{M} &= \mathbf{F}^{M,0}(\boldsymbol{\phi}^{M}) + \Delta\mathbf{F}^{M}, \\ \boldsymbol{\phi}^{M} &= \boldsymbol{\phi}^{M,0} + \Delta\boldsymbol{\phi}^{M}, \end{aligned} \tag{17}$$

where $\Delta\mathbf{F}^{M}$ represents measurement errors, $\boldsymbol{\phi}^{M,0}$ a nominal value for $\boldsymbol{\phi}^{M}$ and $\Delta\boldsymbol{\phi}^{M}$ some physical (manufacturing, operating) tolerances.

For simulation purposes, we may consider a realistic representation of the hierarchy of possible models as

$$\begin{aligned} \mathbf{F}^{H} &= \mathbf{F}^{H,0}(\mathbf{F}^{L}) + \Delta\mathbf{F}^{H}, \\ \mathbf{F}^{L} &= \mathbf{F}^{L,0}(\boldsymbol{\phi}^{H}) + \Delta\mathbf{F}^{L}, \\ \boldsymbol{\phi}^{H} &= \boldsymbol{\phi}^{H,0}(\boldsymbol{\phi}^{L}) + \Delta\boldsymbol{\phi}^{H}, \\ \boldsymbol{\phi}^{L} &= \boldsymbol{\phi}^{L,0} + \Delta\boldsymbol{\phi}^{L}. \end{aligned} \tag{18}$$

where $\boldsymbol{\phi}^{L,0}$, $\boldsymbol{\phi}^{H,0}$, $\mathbf{F}^{L,0}$ and $\mathbf{F}^{H,0}$ are nominal models applicable at different levels. $\Delta\boldsymbol{\phi}^{L}$, $\Delta\boldsymbol{\phi}^{H}$, $\Delta\mathbf{F}^{L}$ and $\Delta\mathbf{F}^{H}$ represent uncertainties or inaccuracies associated with the respective models. $\Delta\boldsymbol{\phi}^{L}$ corresponds to the tolerances $\Delta\boldsymbol{\phi}^{M}$. $\Delta\boldsymbol{\phi}^{H}$ may be due to the approximate nature of an empirical formula. Parasitic effects which are not adequately modeled in $\boldsymbol{\phi}^{H}$ will contribute to $\Delta\mathbf{F}^{L}$, and finally we attribute anything else that causes a mismatch between $\mathbf{F}^{H,0}$ and $\mathbf{F}^{M,0}$ to $\Delta\mathbf{F}^{H}$.

These concepts can be illustrated by the one-section stripline transformer example [30] which we have considered in Section II. Tolerances may be imposed on the physical

parameters including the strip widths and thicknesses, the dielectric constants, the section length and substrate thicknesses (see Fig. 1). Such tolerances correspond to $\Delta\phi^M$ and are represented in the model by $\Delta\phi^L$. We may also use $\Delta\phi^H$ to represent uncertainties associated with the empirical formulas which relate the physical parameters to the equivalent circuit parameters (the effective linewidths, the junction inductances and the effective section length). Mismatches in the terminations at different frequencies may be estimated by ΔF^H (F^H being the actual reflection coefficient; see [30] for more details).

The distinction between different levels of model uncertainties can be quite subtle. As an example, consider the parasitic resistance r associated with an inductor whose inductance is L. Both L and r are functions of the number of turns of a coil (which is a physical parameter). Depending on whether or not r is modeled by the equivalent circuit (i.e., whether or not r is included in ϕ^H), the uncertainty associated with r may appear in $\Delta\phi^H$ or in ΔF^L.

When such uncertainties are present, a single nominal model often fails to represent satisfactorily the physical reality. One effective solution to the problem is to simultaneously consider multiple circuits. We discuss the consequences for design and modeling separately.

Multi-Circuit Design

Our primary concern is to improve production yield and reduce cost in the presence of tolerances $\Delta\phi^L$ and model uncertainties $\Delta\phi^H$. First of all, we represent a realistic situation by multiple circuits as

$$\phi^k = \phi^0 + s^k, \qquad k = 1, 2, \ldots, K, \tag{19}$$

where ϕ^0, ϕ^k and s^k are generic notation for the nominal parameters, the kth set of parameters and a deviate due to the uncertainties, respectively. A more elaborate definition is developed as we proceed.

For each circuit, we define an acceptance index by

$$I_a(\phi) = \begin{cases} 1, & \text{if } H(e(\phi)) \le 0 \\ 0, & \text{otherwise} \end{cases}, \tag{20}$$

where $H(e) \leq 0$, defined in (13), indicates satisfaction of the specifications by ϕ. An estimate of the yield is given by the percentage of acceptable samples out of the total, as

$$Y \simeq \left[\sum_{k=1}^{K} I_a(\phi^k) \right] / K. \tag{21}$$

The merit of a design can then be judged more realistically according to the yield it promises, as illustrated in Fig. 9. Now we shall have a closer look at the definition of multiple circuits.

In the Monte Carlo method the deviates s^k are constructed by generating random numbers using a physical process or arithmetical algorithms. Typically, we assume a statistical distribution for $\Delta\phi^L$, denoted by $D^L(\varepsilon^L)$ where ε^L is a vector of tolerance variables. For example, we may consider a multidimensional uniform distribution on $[-\varepsilon^L, \varepsilon^L]$. Similarly, we assume a $D^H(\varepsilon^H)$ for $\Delta\phi^H$. The uniform and Gaussian (normal) distributions are illustrated in Fig. 10.

At the low level, consider

$$\phi^{L,k} = \phi^{L,0} + s^{L,k}, \qquad k = 1, 2, \ldots, K^L, \tag{22}$$

where $s^{L,k}$ are samples from D^L. At the higher level, we have, for each k,

$$\phi^{H,k,i} = \phi^{H,0} + s^{H,k,i}, \qquad i = 1, 2, \ldots, K^H, \tag{23}$$

where

$$\begin{aligned} \phi^{H,0} &= \phi^{H,0}(\phi^{L,0}), \\ s^{H,k,i} &= \phi^{H,0}(\phi^{L,k}) - \phi^{H,0}(\phi^{L,0}) + \delta^{k,i} \end{aligned} \tag{24}$$

with $\delta^{k,i}$ being samples from D^H.

One might propose a distribution for $s^{H,k,i}$ which presumably encompasses the effect of distribution D^L and distribution D^H. But, while we may reasonably assume simple and independent distributions for $\Delta\phi^L$ and $\Delta\phi^H$, the compound distribution is likely to be complicated and correlated and, therefore, much less desirable.

Centering, Tolerancing and Tuning

Again, in order to simplify the notation, we use ϕ^0 for the nominal circuit and ε for the tolerance variables.

An important problem involves design centering with fixed tolerances, usually relative to corresponding nominal values. We call this the fixed tolerance problem (FTP). The optimization variables are elements of Φ^0, the elements of ε are constant or dependent on the variables, and the objective is to improve the yield.

Since imposing tight tolerances on the parameters will increase the cost of the component fabrication or process operation, we may attempt to maximize the allowable tolerances subject to an acceptable yield. In this case both Φ^0 and ε may be considered as variables. Such a problem is referred to as optimal tolerancing, optimal tolerance assignment, or the variable tolerance problem (VTP). Incidentally, the nominal optimization problem, i.e., the traditional design problem, is sometimes referred to as the zero tolerance problem (ZTP).

Tuning some components of Φ^M after production, whether by the manufacturer or by a customer, is quite commonly used as a means of improving the yield. This process can also be simulated using the model by introducing a vector of designable tuning adjustments τ^k for each circuit, as

$$\Phi^k = \Phi^0 + s^k + \tau^k, \qquad k = 1, 2, ..., K. \tag{25}$$

We have to determine, through optimization, the value of τ^k such that the specifications will be satisfied at Φ^k which may otherwise be unacceptable, as depicted in Figs. 11 and 12. The introduction of tuning, on the other hand, also increases design complexity and manufacturing cost. We seek a suitable compromise by solving an optimization problem in which τ^k are treated as part of the variables.

From nominal design, centering, optimal tolerancing to optimal tuning, we have defined a range of problems which lead to increasingly improved yield but, on the other hand, correspond to increasing complexity. Some specific formulations are discussed in Section V. Analogously to ZTP, FTP and VTP we can define zero tuning, fixed tuning and variable tuning problems [25].

Multi-Circuit Modeling

The uncertainties that affect circuit modeling can be discussed under the following categories.

1) Measurement errors will inevitably exist in practice, as represented by $\Delta \mathbf{F}^M$ in (17): $\mathbf{F}^M = \mathbf{F}^{M,0}(\boldsymbol{\phi}^M) + \Delta \mathbf{F}^M$.

2) Even without measurement errors, the calculated response $\mathbf{F}^{H,0}$ may never be able to match $\mathbf{F}^{M,0}$ perfectly, due to, for example, the use of a model of insufficient order or inadequate complexity. Such an inherent mismatch is accounted for in (18) by $\mathbf{F}^H = \mathbf{F}^{H,0} + \Delta \mathbf{F}^H$.

3) Even if neither $\Delta \mathbf{F}^M$ nor $\Delta \mathbf{F}^H$ exists so that $\mathbf{F}^{H,0} = \mathbf{F}^M$, we may still not be able to uniquely identify $\boldsymbol{\phi}$ from the set of measurements that has been selected. This happens when the system of (generally nonlinear) equations $\mathbf{F}^{H,0}(\boldsymbol{\phi}) - \mathbf{F}^M = \mathbf{0}$, where $\mathbf{F}^M$ is the data, is underdetermined. Typically, this problem occurs when, for any reason, many internal nodes are inaccessible to direct measurement. An over-complicated equivalent circuit, including unknown parasitic elements is frequently at the heart of this phenomenon.

4) The parasitic effects that are not adequately modeled by $\boldsymbol{\phi}^H$ contribute to the uncertainty $\Delta \mathbf{F}^L$. This is another source of interference with the modeling process.

First we consider the case in which modeling is applied to obtain a suitable $\boldsymbol{\phi}$ such that $\mathbf{F}^H(\boldsymbol{\phi})$ approximates $\mathbf{F}^M$. The nominal circuit approach may be able to cope with the uncertainties in 1) and 2), and comes up with a $\boldsymbol{\phi}$ which minimizes the errors $\Delta \mathbf{F}^M$ and $\Delta \mathbf{F}^H$ in a certain sense. But it will not be able to overcome the problem of uniqueness. In practice, we are often unable to determine unambiguously the identifiability of a system, because all these uncertainties can be present at the same time. There will be, typically, a family of solutions which produce reasonable and similar matches between the measured and the calculated responses. We can not, therefore, rely on any particular set of parameters.

The approach of multi-circuit modeling by Bandler et al. [16] can be used to overcome these difficulties. Multiple circuits are created by making deliberate adjustments on the physical parameters Φ^M. For example, we can change the biasing conditions for an active device and obtain multiple sets of measurements. By doing so, we introduce perturbations to the model which cause some parameters in Φ to change by an unknown amount. For this approach to be successful, each physical adjustment should produce changes in only a few parameters in Φ.

Although we do not know the changes in Φ quantitatively, it is often possible to identify which model parameters may have been affected by the physical adjustments. Such a qualitative knowledge may be apparent from the definition of the model or it may come from practical experience. In the attempt to process multiple circuits simultaneously, we define those model parameters that are not supposed to change as common variables and, at the same time, allow the others to vary between different circuits. By doing so, we force the solution to exhibit the desired consistency and, therefore, improve the reliability of the result. In other words, from a family of possible solutions we select the one that conforms to the topological constraints. Bandler et al. have shown an example ([16], Section III.A) in which Φ can not be uniquely identified due to inaccessible nodes. The problem was effectively addressed using the multi-circuit approach.

To formulate this mathematically, let

$$\Phi^k = \begin{bmatrix} \Phi_c^k \\ \Phi_a^k \end{bmatrix}, \tag{26}$$

where $\Phi^k{}_c$ contains the common variables and $\Phi^k{}_a$ contains the variables which are allowed to vary between the kth circuit and the reference circuit Φ^0. We then define the optimization variables by

$$\mathbf{x} = \begin{bmatrix} \Phi^0 \\ \Phi_a^1 \\ . \\ . \\ \Phi_a^K \end{bmatrix}, \tag{27}$$

and state the optimization problem as to

$$\underset{\mathbf{x}}{\text{minimize}}\ U(\mathbf{x}) = \|\mathbf{f}\|_p\ , \tag{28}$$

where

$$\mathbf{f} = [\mathbf{e}^T(\Phi^0)\quad \mathbf{e}^T(\Phi^1)\ \ldots\ \mathbf{e}^T(\Phi^K)]^T\ . \tag{29}$$

Although any ℓ_p norm may be used, the unique property of ℓ_1 discussed in detail by Bandler et al. [16] can be exploited to great advantage. The concept of common and independent variables is depicted in Fig. 13.

Now, suppose that we do not have a clear idea about which model parameters may have been affected by the adjustment on Φ^M. In this case, we let

$$\mathbf{x} = \begin{bmatrix} \Phi^0 \\ \Phi^1 \\ . \\ . \\ \Phi^K \end{bmatrix}, \tag{30}$$

and change the objective function to an ℓ_p norm of

$$\mathbf{f} = \begin{bmatrix} \mathbf{e}(\Phi^0) \\ . \\ . \\ \mathbf{e}(\Phi^K) \\ \alpha_1(\Phi^1 - \Phi^0) \\ . \\ . \\ \alpha_K(\Phi^K - \Phi^0) \end{bmatrix}, \tag{31}$$

where $\alpha_1, \alpha_2, \ldots, \alpha_K$ are nonnegative multipliers (weights).

Using this formulation, while minimizing the errors **e**, we penalize the objective function for any deviates between Φ^k and Φ^0, since our only available knowledge is that only a few parameters in Φ^k should have any significant changes. To be effective, an ℓ_1 norm should be used. A similar principle has been successfully applied to the analog circuit fault location problem [11,33].

Another important application of multi-circuit modeling is to create analytical formulas which link the model Φ to the actual physical parameters Φ^M. Such formulas will become extremely useful in guiding an actual production alignment or tuning procedure. A sequence of adjustments on Φ^M can be systematically made and multiple sets of measurements are taken. By nominal circuit optimization, these measurements would be processed separately to obtain a set of static models. In the presence of uncertainties, a single change in Φ^M may seem to cause fluctuations in all the model parameters. Obviously, such results are of very little use. In contrast, multi-circuit modeling is more likely to produce models that are consistent and reliable. Since the measurements are made systematically, it certainly makes sense to process them simultaneously. Actually, the variables need not be equivalent circuit model parameters. They can include coefficients of a proposed formula as well.

The multi-circuit approach can also be applied to model verification. This is typically related to cases where the parasitic uncertainty $\Delta \mathbf{F}^L$ has put the validity of a model in doubt. Instead of defining common and independent variables explicitly, we use the formulation of (30) and (31). If consistent results are obtained, then our confidence in the model is strengthened. Otherwise we should probably reject the current model and consider representing the parasitics more adequately. A convincing example has been demonstrated by Bandler et al. ([16], section V, Test 2).

The commercial packages TOUCHSTONE [118] and SUPER-COMPACT [113] allow a hierarchy of circuit blocks and permit the use of variable labels. Multiple circuits and common variables can be easily defined utilizing these features. ℓ_1 optimiztion, however, is not yet available.

Examples of Multi-circuit Modeling

Our first example is robust modeling of a FET device. Consider the equivalent circuit model shown in Fig. 14 (which has been widely used by commercial packages such as TOUCHSTONE [118] and SUPER-COMPACT [113]). Practical modeling of FET devices has been commonly troubled by nonunique solutions.

In order to demonstrate the multi-circuit modeling approach, we employ three sets of actual measurements on scattering parameters of a FET device which were taken at 17 frequency points from 2GHz to 18GHz, 1GHz apart, under the following biasing conditions [105].

1. $V_{ds} = 4V$, $V_{gs} = 0.00V$, $I_{ds} = 177mA$.
2. $V_{ds} = 4V$, $V_{gs} = -1.74V$, $I_{ds} = 92mA$.
3. $V_{ds} = 4V$, $V_{gs} = -3.10V$, $I_{ds} = 37mA$.

Eleven model parameters, namely $\{R_g, R_d, L_s, \tau, R_{ds}, R_i, R_s, C_{gs}, C_{dg}, C_{ds}, g_m\}$, are taken as variables. The first four parameters are considered to be bias insensitive and, therefore, treated as common variables. Formulas (27)-(29) are used, where K = 3 (three circuits) and ℓ_1 optimization is employed. The match between the model responses and the measurements, at both the starting point and the solution, for the first biasing condition is shown in Fig. 15. Complete details have been reported by Bandler et al. [21].

The second example uses data provided by Com Dev [51]. It is related to establishing an experimental relationship between physical and model parameters (Daijavad [52]). Consider a 6th order multi-cavity filter centered at 11783MHz with a 56MHz bandwidth. The equivalent circuit of Atia and Williams [5] has been described in Section III. Starting from a reference position, three coupling screws, whose positions are represented as elements of ϕ^L, which were assumed to control the couplings M_{12}, M_{34} and M_{56}, i.e., elements of ϕ^H were adjusted. Each one was adjusted four times, twice in the clockwise direction (screw increasingly penetrates the cavity) for 90 and 180 degrees, as well as twice counterclockwise.

After each adjustment, the filter responses (input/output return loss, insertion loss and group delay) were measured. Using formulas (30)-(31), the measurements have been processed simultaneously to identify the filter coupling values. The variation of coupling values versus the relative position of the screws is shown in Fig. 16.

Concluding Remarks

In the foregoing discussions, for each of the problems we have focused on the dominant uncertainties. The statistical design approach has addressed $\Delta\Phi^L$ and $\Delta\Phi^H$. By multi-circuit modeling we try to overcome the difficulties caused by ΔF^M, ΔF^H, ΔF^L and the possible rank deficiency in $F^H(\Phi)$. The complete problem in which all these uncertainties are accounted for has not been formally solved.

Consider the problem of computer-aided actual tuning of an outcome. Here, the physical parameters Φ^M and the measured responses F^M are the fundamental variables and functions, respectively. A plausible scheme involves three phases at each tuning iteration. At the modeling phase, measurements are made from which models at different levels are established. At the design phase a computer program will suggest the necessary tuning adjustments using the models. The implementation phase involves actual tuning by a human operator or through some automated mechanism. The modeling must be robust despite possible uncertainties. The crucial relationship between Φ^H and Φ^L must be constantly updated. Anticipating imprecisions in the implementation, the determination of tuning adjustments must be toleranced. In other words, an implementable algorithm will have to combine the concepts of multi-circuit modeling and design.

V. TECHNIQUES FOR STATISTICAL DESIGN

In Section IV we have generally discussed uncertainties at different levels and, in particular, we have expressed our desire to maximize yield in the presence of uncertainties.

Optimal tolerancing and tuning have also been identified as means to further reduce cost in the actual production.

We begin this section with a review of some existing techniques for statistical design. Some of the earliest work in this area came from Karafin [79], Pinel and Roberts [97], Butler [42], Elias [60], Bandler, Liu and Tromp [29]. During the years, significant contributions have been made by, among others, Director and Hachtel [55] (the simplicial method), Soin and Spence [112] (the gravity method), Bandler and Abdel-Malek [1,2,9] (multi-dimensional approximation), Biernacki and Styblinski [36] (dynamic constraint approximation), Polak and Sangiovanni-Vincentelli [100] (a method using outer approximation), as well as Singhal and Pinel [111] (the parametric sampling method). Following the review, we propose a generalized ℓ_p centering algorithm.

A commonly assumed cost versus yield curve [111] is shown in Fig. 17. Actually, hard data is difficult to obtain and, as we shall see, rather abstract objective functions are often selected for the tolerance-yield design problem. Figure 18 shows a design with a 100% yield and a second design corresponding to the minimum cost.

Worst-case Design

By this approach, we attempt to achieve a 100% yield. Since it means that the specifications have to be satisfied for all the possible outcomes, we need to consider only the worst cases.

Bandler et al. [28,29] have formulated it as a nonlinear programming problem as

$$\begin{aligned} &\underset{\mathbf{x}}{\text{minimize}}\ C(\mathbf{x}) \\ &\text{subject to } \mathbf{e}(\boldsymbol{\phi}^k) \le 0\,, \quad \text{for all } k\,, \end{aligned} \tag{32}$$

where $C(\mathbf{x})$ is a suitable cost function and the points $\boldsymbol{\phi}^k$ are the worst cases. For instance, we may have

$$C(\mathbf{x}) = \sum_{i \in I_\varepsilon} \frac{a_i}{\varepsilon_i} + \sum_{i \in I_t} b_i\, t_i\,, \tag{33}$$

where I_ε and I_t are index sets identifying the toleranced and tunable parameters, respectively. ε_i and t_i are the tolerance and the tuning range, respectively, associated with the ith parameter. a_i and b_i are nonnegative weights. A cost function can also be defined for relative tolerances and tuning by including ϕ^0_i into (33). A critical part of this approach is the determination of the worst cases. Vertices of the tolerance region, for example, are possible candidates for the worst cases by assuming one-dimensional convexity. The yield function does not enter (32) explicitly, instead, a 100% yield is implied by a feasible solution.

Bandler and Charalambous [13] have demonstrated a solution to (32) by minimax optimization. Polak and Sangiovanni-Vincentelli [100] have proposed a different but equivalent formulation which involves a nondifferentiable optimization.

A worst-case design is not always appropriate. While attempting to obtain a 100% yield, the worst-case approach may necessitate unrealistically tight tolerances, or demand excessive tuning. In either case, the cost may be too high. A perfect 100% yield may not even be realizable.

Methods of Approximating the Acceptable Region

Since yield is given by the percentage of model outcomes that fall into the acceptable region, we may wish to find an approximation to that region. The acceptable region has been defined in (16) as $R_a = \{\boldsymbol{\phi} \mid H(\mathbf{e}(\boldsymbol{\phi})) \leq 0\}$.

Director and Hachtel [55] have devised a simplicial approximation approach. It begins by determining points $\boldsymbol{\phi}^k$ on the boundary of R_a which is given by $\Omega_a = \{\boldsymbol{\phi} \mid H(\mathbf{e}(\boldsymbol{\phi})) = 0\}$. The convex hull of these points forms a polyhedron. The largest hypersphere inscribed within the polyhedron gives an approximation to R_a and is found by solving a linear programming problem. Using line searches more points on the boundary are located and the polyhedron is expanded. The process thus provides a monotone increasing lower bound on the yield. The center and radius of the hypersphere can be used to determine

centered nominal point and the tolerances, respectively. The application of this method is, however, severely limited by the assumption of a convex acceptable region.

Bandler and Abdel-Malek [1,2,9] have presented a method which approximates each $e_j(\Phi)$ by a low-order multi-dimensional polynomials. Model simulations are performed at some Φ^k selected around a reference point. From the values of $e_j(\Phi^k)$ the coefficients of the approximating polynomial are determined by solving a linear system of equations. Appropriate linear cuts are constructed to approximate the boundary Ω_a. The yield is estimated through evaluation of the hypervolumes that lie outside R_a but inside the tolerance region. In critical regions these polynomial approximations are updated during optimization. The one-dimensional convexity assumption for this method is much less restrictive than the multi-dimensional convexity required by the simplicial approach. Sensitivities for the estimated yield are also available.

Recently, Biernacki and Styblinski [36] have extended the work on multi-dimensional polynomial approximation by considering a dynamic constraint approximation scheme. It avoids the large number of base points required for a full quadratic interpolation by selecting a maximally flat interpolation. During optimization, whenever a new base point is added the approximation is updated. It shows improved accuracy compared with a linear model as well as reduced computational effort compared with a full quadratic model.

The Gravity Method

Soin and Spence [112] proposed a statistical exploration approach. Based on a Monte Carlo analysis, the centers of gravity of the failed and passed samples are determined as, respectively,

$$\Phi^f = \left[\sum_{k \in J} \Phi^k \right] / K_{fail} ,$$
$$\Phi^p = \left[\sum_{k \notin J} \Phi^k \right] / K_{pass} , \qquad (34)$$

where J is the index set identifying the failed samples. K_{fail} and K_{pass} are the numbers of failed and passed samples, respectively. The nominal point ϕ^0 is then adjusted along the direction $\mathbf{s} = \phi^p - \phi^f$ using a line search. This algorithm is simple but also heuristic. It is not clear as how the gravity centers are related to the yield in a general multi-dimensional problem.

The Parametric Sampling Method

The parametric sampling approach by Singhal and Pinel [111] has provided another promising direction. A continuous estimate of yield (as opposed to the Monte Carlo estimate using discrete samples) is given by the following integral

$$Y(\mathbf{x}) = \int_{-\infty}^{+\infty} I_a(\phi)\, \Gamma(\phi, \mathbf{x})\, d\phi \ , \tag{35}$$

where $I_a(\phi)$ is the acceptance index defined in (20) and $\Gamma(\phi,\mathbf{x})$ the parameter distribution density function which depends on the design variables $\mathbf{x}$ (e.g., the nominal point specifies the mean value and the tolerances control the standard deviations). Normally, in order to estimate the yield, we generate samples ϕ^k, k = 1, 2,..., K, from the component density Γ, perform K circuit analyses and then take the average of $I_a(\phi^k)$. For each new set of variables $\mathbf{x}$ we would have a new density function and, therefore, the sampling and circuit analyses have to be repeated.

The approach of parametric sampling is based on the concept of importance sampling as

$$Y(\mathbf{x}) = \int_{-\infty}^{+\infty} I_a(\phi)\, \frac{\Gamma(\phi,\mathbf{x})}{h(\phi)}\, h(\phi)\, d\phi \ , \tag{36}$$

where $h(\phi)$ is called the sampling density function. The samples ϕ^k are generated from $h(\phi)$ instead of $\Gamma(\phi,\mathbf{x})$. An estimate of the yield is made as

$$Y(\mathbf{x}) \simeq \frac{1}{K} \sum_{k=1}^{K} I_a(\phi^k)\, \frac{\Gamma(\phi^k,\mathbf{x})}{h(\phi^k)} = \frac{1}{K} \sum_{k=1}^{K} I_a(\phi^k)\, W(\phi^k, \mathbf{x}) \, . \tag{37}$$

The weights $W(\phi^k,\mathbf{x})$ compensate for the use of a sampling density different from the component density.

This approach has two clear advantages. Firstly, once the indices $I_a(\phi^k)$ are calculated, no more model simulations are required when $\mathbf{x}$ is changed. Furthermore, if Γ is a differentiable density function, then gradients of the estimated yield are readily available. Hence, powerful optimization techniques may be employed. In practice the algorithm starts with a large number of base points sampled from $h(\phi)$ to construct the initial databank. To maintain a sufficient accurary, the databank needs to be updated by adding new samples during optimization.

This approach, however, can not be applied to non-differentiable density functions such as uniform, discrete and truncated distributions. It can be extended to include some tunable parameters if the tuning ranges are fixed or practically unlimited. In this case the acceptance index $I_a(\phi^k)$ is defined as 1 if ϕ^k is acceptable after tuning. If ϕ^k is unacceptable before tuning, then whether it can be tuned and, if so, by how much may have to be determined through optimization. Variable tuning range (in order to minimize cost) can not be accommodated by the parametric sampling method.

Generalized ℓ_p Centering

Here, we propose a generalized ℓ_p centering algorithm which encompasses, in a unified formulation, problems of 100% yield (worst-case design) and less than 100% yield.

First we consider the centering problem where we have fixed tolerances and no tuning. Only the nominal point ϕ^0 is to be optimized. We can achieve a worst-case minimax design by

$$\underset{\mathbf{x}}{\text{minimize}}\ U(\mathbf{x}) = H_\infty\left(\begin{bmatrix} e(\phi^1) \\ . \\ . \\ e(\phi^K) \end{bmatrix}\right) = \max_k \max_j \{e_j(\phi^k)\}, \tag{38}$$

where the multiple circuits ϕ^k are related to ϕ^0 according to (19).

If a 100% yield is not attainable, we would naturally look for a solution where the specifications are met by as many points (out of K circuits) as possible. For this purpose minimax is not a proper choice, since unless and until the worst case is dealt with nothing else seems to matter. An apparent remedy is to use a generalized ℓ_2 or ℓ_1 function (i.e., $H_2(.)$ or $H_1(.)$) instead of H_∞ in (38). However, even with such a modification, in (38) each circuit is represented by a set of error functions and the meaning of such a representation is not altogether clear.

A better approach to the problem is to find, for each ϕ^k, a scalar function which will indicate directly whether ϕ^k satisfies or violates the specifications and by how much. For this purpose, we choose a set of generalized ℓ_p functions as

$$v_k(\mathbf{x}) = H_p(e(\phi^k)), \qquad k = 1, 2, \ldots, K. \tag{39}$$

The sign of v_k indicates the acceptability of ϕ^k while the magnitude of v_k measures, so to speak, the distance between ϕ^k and the boundary of the acceptable region. For example, with $p = \infty$ the distance is measured in the worst-case sense whereas for $p = 2$ it will be closer to a Euclidean norm.

We can define a generalized ℓ_p centering as

$$\underset{\mathbf{x}}{\text{minimize}}\ U(\mathbf{x}) = H_p(\mathbf{u}(\mathbf{x})), \tag{40}$$

where

$$\mathbf{u}(\mathbf{x}) = \begin{bmatrix} a_1 v_1 \\ \cdot \\ \cdot \\ a_K v_K \end{bmatrix} = \begin{bmatrix} a_1 H_q(e(\phi^1)) \\ \cdot \\ \cdot \\ a_K H_q(e(\phi^K)) \end{bmatrix} \tag{41}$$

and $a_1, a_2, \ldots, a_K$ are a set of positive multipliers. With different p and q it leads to a variety of algorithms for yield enchancement. We discuss separately the case where a nonpositive $U(\mathbf{x})$ exists and the case where we always have $U(\mathbf{x}) > 0$.

In the first case, the existence of a $U(\mathbf{x}) \leq 0$ indicates that a 100% yield is attainable. We should point out that for a given $\mathbf{x}$ the sign of $U(\mathbf{x})$ does not depend on p, q or any α_k. However, the optimal solution $\mathbf{x}$ at which $U(\mathbf{x})$ attains its minimum is dependent on p, q and $\boldsymbol{\alpha}$. This means that using any values of p, q and $\boldsymbol{\alpha}$ we will be able to achieve a $U(\mathbf{x}) \leq 0$ (i.e., to achieve a 100% yield). Furthermore, by using different p, q and $\boldsymbol{\alpha}$, we influence the centering of $\boldsymbol{\phi}^0$. Interestingly, the worst-case centering (38) becomes a special case by letting both $p, q = \infty$ and using unit multipliers.

Now consider the case where the optimal yield is less than 100%. In this case we propose the use of $p = 1$ and $q = 1$ in (40). Also, given a starting point $\mathbf{x}_0$, we define the set of multipliers by

$$\alpha_k = 1/|v_k(\mathbf{x}_0)|, \quad k = 1, 2, \ldots, K \tag{42}$$

Our proposition is based on the following reasoning (a more complete theoretical justification is reserved for a future paper).

Consider the ℓ_p sum given by

$$\sum_{k \in J} [u_k(\mathbf{x})]^p \tag{43}$$

where $J = \{k \mid u_k > 0\}$. As $p \rightarrow 0$ (43) approaches the total number of unacceptable circuits which we wish to minimize. The smallest p that gives a convex approximation is 1. This leads to the generalized ℓ_1 objective function given by

$$U(\mathbf{x}) = \sum_{k \in J} u_k(\mathbf{x}) = \sum_{k \in J} \alpha_k v_k(\mathbf{x}). \tag{44}$$

With the multipliers defined by (42), the value of the objective function at the starting point, namely $U(\mathbf{x}_0)$, is precisely the count of unacceptable circuits. Also, notice that the magnitude of v_k measures the closeness of $\boldsymbol{\phi}^k$ to the acceptable region. A small $|v_k|$ indicates that $\boldsymbol{\phi}^k$ is close to be satisfying or violating the specifications. Therefore, we assign a large multiplier to it so that more emphasis will be given to $\boldsymbol{\phi}^k$ during optimization. On the other hand, we deemphasize those points that are far away from the boundary of the acceptable region because their contributions to the yield are less likely to change.

A sequential process may also be constructed in which we solve (40), update the multipliers at the solution and repeat the optimization. Bearing in mind the fact that yield optimization should always be initiated from a good nominal design and, therefore, drastic changes in the variables are unlikely to take place, we can expect a stable process of updating the multipliers.

The objective function $U(\mathbf{x})$ in (40) can be rewritten, when $p = q = 1$ and $U(\mathbf{x}) > 0$, as $U(\mathbf{x}) = H_1{}^+(\mathbf{f})$, where

$$\mathbf{f} = [a_1 e^T(\phi^1) \quad a_2 e^T(\phi^2) \quad \ldots \quad a_K e^T(\phi^K)]^T. \tag{45}$$

On the other hand, when it is negative, $U(\mathbf{x})$ is a continuously differentiable function.

Optimal Tolerancing and Tuning

The generalized ℓ_p centering algorithm can be extended to accommodate considerations of tolerances and tuning.

We need to define a function or functions which appropriately relate the tolerances and tuning to the cost of production. One possible choice of such a cost function is, similar to (33),

$$C(\mathbf{x}) = \sum_{i \in I_\varepsilon} \frac{a_i}{\varepsilon_i} + \sum_{i \in I_t} b_i t_i - C_0, \tag{46}$$

where I_ε and I_t are index sets identifying the toleranced and tunable parameters, respectively. a_i and b_i are nonnegative weights. C_0 is a realistic target for the cost. It is also possible to define several cost functions representing separate tradeoffs in a complex environment such as in the design of integrated circuits (similarly to the multiple objectives described by Brayton et al. [38]). For example, (46) may be broken up into several cost functions associated with separate groups of parameters.

By this formulation, the cost function is treated in the same way as we treat $e(\phi)$. When $C(\mathbf{x}) \leq 0$, we say that $\mathbf{x}$ satisfies the specification on the cost (as given by C_0). In fact,

we can adjust C_0 and the weights a_i, b_i in (46) such that $C(\mathbf{x})$ is made comparable in value with the error functions.

The set of optimization variables $\mathbf{x}$ should include the nominal circuit parameters ϕ^0, the tolerances ε_i, $i \in I_\varepsilon$, and the tuning adjustments τ^k_i, $|\tau^k_i| \leq t_i$, $i \in I_t$, $k = 1, 2, \ldots, K$, for the K circuits defined by (25).

In this way, the approach of generalized ℓ_p centering can be applied directly as

$$\underset{\mathbf{x}}{\text{minimize}}\ U(\mathbf{x}) = H_p(\mathbf{u}(\mathbf{x}))\,, \tag{47}$$

where

$$\mathbf{u}(\mathbf{x}) = \begin{bmatrix} a_1 v_1 \\ \cdot \\ \cdot \\ a_K v_K \\ a_{K+1} v_{K+1} \end{bmatrix} = \begin{bmatrix} a_1 H_q(e(\phi^1)) \\ \cdot \\ \cdot \\ a_K H_q(e(\phi^K)) \\ a_{K+1} C(\mathbf{x}) \end{bmatrix}. \tag{48}$$

Following the discussions of the last subsection, we should be able to obtain a solution, say $\mathbf{x}^*$, for (47). If $U(\mathbf{x}^*) \leq 0$ then the design specifications have all been satisfied and the target for cost has been met. If, on the other hand, $U(x^*) > 0$, we can conclude that either the design specifications are too tight or the target cost is unrealistic or perhaps both.

One important feature of the ℓ_p centering approach is its capability of accommodating arbitrary tolerance distributions at different levels ($\Delta\phi^L$, $\Delta\phi^H$ or both), because they only influence the generation of ϕ^k. It provides a unified formulation for the various statistical design problems (worst-case, yield enhancement, tolerancing, tuning, etc.). In fact, the similarity between the formulas for ℓ_p centering and those for multi-circuit modeling, as appeared in Section IV, suggests that they belong to the same family of multi-circuit optimization approaches.

VI. EXAMPLES OF STATISTICAL DESIGN

Example 1

The classical two-section 10:1 transmission line transformer, originally proposed by Bandler et al. [28] to test minimax optimizers, is a good example to illustrate graphically the basic ideas of centering and tolerancing. An upper specification on the reflection coefficient as $|\rho| \leq 0.55$ and 11 frequencies {0.5, 0.6,..., 1.5 GHz} are considered. The lengths of the transmission lines are fixed at the quarter-wave length while the characteristic impedances Z_1 and Z_2 are to be toleranced and optimized. Fig. 19 shows the minimax contours, the minimax nominal solution, as well as the worst-case solutions [28] for

$$\text{P0:} \quad \begin{aligned} &\text{minimize } C_1 = Z_1^0/\varepsilon_1 + Z_2^0/\varepsilon_2 \\ &\text{subject to } Y = 100\% \end{aligned},$$

$$\text{P1:} \quad \begin{aligned} &\text{minimize } C_2 = 1/\varepsilon_1 + 1/\varepsilon_2 \\ &\text{subject to } Y = 100\% \end{aligned},$$

where ε_1, ε_2 denote tolerances on Z_1 and Z_2 (assuming independent uniform distributions), and Y is the yield. The cost functions C_1 and C_2 correspond to, respectively, relative and absolute tolerancing problems.

Another two problems of less than 100% yield have been considered by Bandler and Abdel-Malek [9] as

P2: minimize C_2 subject to $Y \geq 90\%$,

P3: minimize C_2/Y.

The optimal tolerance regions and nominal values for P2 and P3 are shown in Fig. 20. For more details see the original paper [9].

Example 2

The example due to Singhal and Pinel [111] of statistical design of a Chebyshev lowpass filter is used as the second example. The circuit is shown in Fig. 21. Fifty-one frequencies {0.02, 0.04,..., 1.0, 1.3 Hz} are considered. An upper specification of 0.32 dB on the insertion loss is defined for frequencies from 0.02 to 1.0 Hz. A lower specification of 52 dB on the insertion loss is defined at 1.3 Hz. A nominal design obtained by standard filter synthesis is given in Table I.

For statistical design the 11 components are assumed to be independently normally distributed. A cost function of

$$C = \left\{ \sum_{i=1}^{11} (\phi_i^0/\varepsilon_i) \right\} / Y$$

is considered, where ϕ_i^0 is the mean value (nominal value), $\varepsilon_i = 3\sigma_i$ with σ_i being the standard deviation, and Y is the yield.

A worst-case design and a minimum cost design were reported in [111] and reproduced in Table I. As one can see, the worst-case solution gives a perfect (100%) yield but a rather high cost (18.68). The minimum cost design was obtained using the parametric sampling techniques [111] with two design cycles starting from the worst-case solution. A total of 800 circuit analyses were required to construct the database. The total CPU time required was reported to be 75 seconds on the IBM 370/158. This excludes the cost of worst-case design and yield verification by Monte Carlo analyses.

In an interesting arrangement by Wehrhahn and Spence [122] the performance of seven design centering algorithms was compared. The filter used here was one of the test problems.

Example 3

As we have pointed out earlier in this paper, the parametric sampling method can not be applied to non-differentiable (such as uniform) distributions. Our third example considers

the same filter as in Example 2 but assumes a uniformly distributed 1.5% relative tolerance for each component. The generalized ℓ_p centering algorithm described in section V is used with $p = 1$. The nominal solution by standard synthesis was used as starting point which has a 49% yield (w.r.t. the tolerances specified). A 84% yield is achieved at the solution which involves a sequence of three design cycles with a total CPU time of 66 seconds on the VAX 8600. Some details are provided in Table II.

Discussion

In relation to the generalized ℓ_p centering, we should point out that as long as the yield is less than 100%, negative error functions do not contribute to the generalized ℓ_p objective. Therefore, under certain conditions, the circuit analyses corresponding to these error functions can be saved. Such a saving becomes even more significant when a higher yield is achieved (since more functions become negative). A suitable technique has been incorporated in solving Example 3. A similar concept can be found in worst-case design where we consider candidates for the worst case only.

Another point worth making is that a statistical design should always be initiated from a good nominal solution. Any attempt to demonstrate yield optimization starting from an unacceptable nominal point (i.e., from a very low yield) has no practical value.

VII. GRADIENT-BASED OPTIMIZATION METHODS

So far we have concentrated on translating our practical concerns into mathematical expressions. Now we turn our attention to the solution methods for optimization problems.

The studies in the last two decades on the theoretical and algorithmic aspects of optimization techniques have produced a great deal of results. Modern state-of-the-art methods have largely replaced the primitive trial-and-error approach. Especially, gradient-based optimization methods have gained increasing popularity in recent years for their

effectiveness and efficiency. They demonstrate in general a far superior performance to the random or direct methods.

The majority of gradient-based methods belong to the Gauss-Newton, quasi-Newton and conjugate gradient families. All these are iterative algorithms which, from a given starting point $\mathbf{x}_0$, generate a sequence of points $\{\mathbf{x}_k\}$. The success of an algorithm depends on whether $\{\mathbf{x}_k\}$ will converge to a point $\mathbf{x}^*$ and if so, whether $\mathbf{x}^*$ will be a stationary point. An iterative algorithm is described largely by one of its iterations as how to obtain $\mathbf{x}_{k+1}$ from $\mathbf{x}_k$.

The following notation is used: $\mathbf{x}$ for a set of n variables, $U(\mathbf{x})$ for the objective function and ∇U for the gradient vector of U. When $U(\mathbf{x})$ is defined by an ℓ_p function, we use $\mathbf{f}$ to denote the set of m individual functions so that $U = H(\mathbf{f})$. The first-order derivatives of a function f_j are written as

$$\mathbf{f}_j' = \left[\frac{\partial f_j}{\partial x_1} \quad \frac{\partial f_j}{\partial x_2} \cdots \frac{\partial f_j}{\partial x_n} \right]^T, \tag{48}$$

and the Jacobian matrix of $\mathbf{f}$ as

$$\mathbf{G} = [\mathbf{f}_1' \ \mathbf{f}_2' \cdots \mathbf{f}_m']^T. \tag{49}$$

<u>ℓ_p Optimization and Mathematical Programming</u>

ℓ_1, ℓ_2 and ℓ_∞ are the most distinctive and by far the most useful members of the ℓ_p family. Apart from their unique theoretical properties, it is very important from the algorithmic point of view that linear ℓ_1, ℓ_2 and ℓ_∞ problems can be solved exactly using linear or quadratic programming techniques. Besides, all the other members of the ℓ_p family have a continuously differentiable objective function and, therefore, can be treated similarly to the ℓ_2 case.

An ℓ_1, ℓ_2 or ℓ_∞ optimization problem can be converted into a mathematical program. The concepts of local linearization and optimality conditions are often clarified by the equivalent formulation.

For instance, the minimization of $\|\mathbf{f}\|_1$ is equivalent to

$$\underset{\mathbf{x},\mathbf{y}}{\text{minimize}} \quad \sum_{j=1}^{m} y_j \tag{50}$$

subject to

$$y_j \geq f_j(\mathbf{x}), \quad y_j \geq -f_j(\mathbf{x}), \quad j = 1, 2, \ldots, m.$$

The one-sided ℓ_1 problem can be treated as

$$\underset{\mathbf{x},\mathbf{y}}{\text{minimize}} \quad \sum_{j=1}^{m} y_j \tag{51}$$

subject to

$$y_j \geq f_j(\mathbf{x}), \quad y_j \geq 0, \quad j = 1, 2, \ldots, m.$$

Other equivalent formulations are summarized in Table III. For the convenience of presentation, we denote these mathematical programming problems by $P(\mathbf{x},\mathbf{f})$. One important feature of $P(\mathbf{x},\mathbf{f})$ is that it has a linear or quadratic objective function. If $\mathbf{f}$ is a set of linear functions, then $P(\mathbf{x},\mathbf{f})$ becomes a linear or quadratic program which can be solved using standard techniques. Equally importantly, linear constraints can be easily incorporated into the problem. Let $P(\mathbf{x},\mathbf{f},D)$ be the problem of $P(\mathbf{x},\mathbf{f})$ subject to a set of linear constraints of the form

$$D: \quad \begin{aligned} &\mathbf{a}_\ell^T \mathbf{x} + b_\ell = 0, \quad \ell = 1, 2, \ldots, L_{eq}, \\ &\mathbf{a}_\ell^T \mathbf{x} + b_\ell \geq 0 \quad \ell = L_{eq} + 1, \ldots, L, \end{aligned} \tag{52}$$

where $\mathbf{a}_\ell$ and b_ℓ are constants. If $P(\mathbf{x},\mathbf{f})$ is a linear or quadratic program, so is $P(\mathbf{x},\mathbf{f},D)$. In other words, unconstrained and linearly constrained linear ℓ_1, ℓ_2 and ℓ_∞ problems can be solved using standard linear or quadratic programming techniques.

Gauss-Newton Methods Using Trust Regions

For a general problem, we may, at each iteration, substitute $\mathbf{f}$ with a linearized model $\bar{\mathbf{f}}$ so that $P(\mathbf{x},\bar{\mathbf{f}})$ can be solved.

For a Gauss-Newton type method, at a given point $\mathbf{x}_k$, a linearization of $\mathbf{f}$ is made as

$$\bar{\mathbf{f}}(\mathbf{h}) = \mathbf{f}(\mathbf{x}_k) + \mathbf{G}(\mathbf{x}_k)\,\mathbf{h}, \tag{53}$$

where G is the Jacobian defined in (49). We then solve the linear or quadratic program $P(\mathbf{h},\bar{\mathbf{f}},D)$, where

$$D: \quad \begin{aligned} \Lambda_k &\geq h_j, \quad i = 1,2,\ldots,n, \\ \Lambda_k &\geq -h_j, \quad i = 1,2,\ldots,n. \end{aligned} \tag{54}$$

These additional constraints define a trust region in which the linearized model $\bar{\mathbf{f}}$ is believed to be a good approximation to $\mathbf{f}$.

Another way to look at it is that we have applied a semi-linearization (Madsen [88]) to $U(\mathbf{x}) = H(\mathbf{f})$ resulting in

$$\bar{U}(\mathbf{h}) = H(\bar{\mathbf{f}}(\mathbf{h})). \tag{55}$$

It is important to point out that (55) is quite different from a normal linearization as $U(\mathbf{h}) \simeq U(\mathbf{x}_k) + [\nabla U(\mathbf{x}_k)]^T\mathbf{h}$ which corresponds to a steepest descent method. In fact ∇U may not even exist.

Denote the solution of $P(\mathbf{h},\bar{\mathbf{f}},D)$ by $\mathbf{h}_k$. If $\mathbf{x}_k+\mathbf{h}_k$ reduces the original objective function we take it as the next iterate, i.e., if $U(\mathbf{x}_k+\mathbf{h}_k) < U(\mathbf{x}_k)$ then $\mathbf{x}_{k+1} = \mathbf{x}_k+\mathbf{h}_k$. Otherwise we let $\mathbf{x}_{k+1} = \mathbf{x}_k$. In the latter case, the trust region is apparently too large and, consequently, should be reduced.

Actually, the local bound Λ_k in (54) is adjusted according to the goodness of the linearized model. More precisely, if

$$U(\mathbf{x}_k) - U(\mathbf{x}_k + \mathbf{h}_k) \leq \delta_1[U(\mathbf{x}_k) - \bar{U}(\mathbf{h}_k)], \tag{56}$$

then the trust region appears to be too large and the bound is decreased: $\Lambda_{k+1} = K_1\Lambda_k$. Otherwise, if

$$U(\mathbf{x}_k) - U(\mathbf{x}_k + \mathbf{h}_k) \geq \delta_2[U(\mathbf{x}_k) - \bar{U}(\mathbf{h}_k)], \tag{57}$$

then the bound is increased: $\Lambda_{k+1} = K_2\Lambda_k$. If neither (56) nor (57) holds then $\Lambda_{k+1} = \Lambda_k$. The constants $\{\delta_1, \delta_2, K_1, K_2\}$ should satisfy $0 < \delta_1 < \delta_2 < 1$ and $0 < K_1 < 1 < K_2$. {0.25, 0.75, 0.25, 2} have been used in some implementations [26,27,75,76].

The above describes the essence of a class of algorithms due to Madsen who has called it Method 1. Madsen [88] has shown that the algorithm provides global convergence in which

the proper use of trust regions constitutes a critical part. In some other earlier work by Osborne and Watson [95,96] the problem $P(\mathbf{h},\bar{\mathbf{f}})$ was solved without incorporating a trust region and the solution $\mathbf{h}_k$ was used as the direction for a line search. For their methods no convergence can be guaranteed and $\{\mathbf{x}_k\}$ may even converge to a non-stationary point.

Normally for the least-squares we have to solve a quadratic program at each iteration, which can be a time-consuming process. A remarkable alternative is the Levenberg-Marquardt [86,91] method. Given $\mathbf{x}_k$, it solves

$$\underset{\mathbf{h}}{\text{minimize}}\ \mathbf{h}^T(\mathbf{G}^T\mathbf{G} + \theta_k \mathbf{1})\mathbf{h} + 2\mathbf{f}^T\mathbf{G}\mathbf{h} + \mathbf{f}^T\mathbf{f}\ , \tag{58}$$

where $\mathbf{G} = \mathbf{G}(\mathbf{x}_k)$, $\mathbf{f} = \mathbf{f}(\mathbf{x}_k)$ and $\mathbf{1}$ is an identity matrix. The minimizer $\mathbf{h}_k$ is obtained simply by solving the linear system

$$(\mathbf{G}^T\mathbf{G} + \theta_k \mathbf{1})\,\mathbf{h}_k = \mathbf{G}^T\mathbf{f} \tag{59}$$

using, for example, LU factorization. The Levenberg-Marquardt parameter θ_k is very critical for this method. First of all, it is made to guarantee the positive definiteness of (59). Furthermore, it plays, roughly speaking, an inversed role of Λ_k to control the size of a trust region. When $\theta_k \to \infty$, $\mathbf{h}_k$ gives an infinitesimal steepest descent step. When $\theta_k = 0$, $\mathbf{h}_k$ becomes the solution to $P(\mathbf{h},\bar{\mathbf{f}})$ without bounds, which is equivalent to having $\Lambda_k \to \infty$. Therefore, the rules for updating θ_k should be opposite to those for Λ_k.

The Gauss-Newton method using trust regions has been implemented as an important element in the minimax and ℓ_1 algorithms of Hald and Madsen [75,76]. Also, the concept of trust region has been discussed in a broader context by Moré in a recent survey [92].

Quasi-Newton methods

Quasi-Newton methods (also known as variable metric methods) are originated in and steadily upgraded from the work of Davidon [53], Broyden [39,40] as well as Fletcher and Powell [65].

For a differentiable $U(\mathbf{x})$, a quasi-Newton step is given by

$$\mathbf{h}_k = -\alpha_k \mathbf{B}_k^{-1} \nabla U(\mathbf{x}_k), \tag{60}$$

where $\mathbf{B}_k$ is an approximation to the Hessian of $U(\mathbf{x})$ and the step size controlling parameter α_k is to be determined through a line search. However, on some occasions such as in the ℓ_1 or minimax case, the gradient ∇U may not exist, much less the Hessian.

We can gain more insight to the general case by examining the optimality conditions. Applying the Kuhn-Tucker conditions for nonlinear programming [80] to the equivalent problem $P(\mathbf{x},\mathbf{f})$, we shall find a set of optimality equations

$$\mathbf{R}(\mathbf{x}) = \mathbf{0}. \tag{61}$$

Since a local optimum $\mathbf{x}^*$ must satisfy these equations, we are naturally motivated to solve (61), as a means of finding the minimizer of $U(\mathbf{x})$. A quasi-Newton step for solving nonlinear equations (61) is given by

$$\mathbf{h}_k = -\alpha_k \mathbf{J}_k^{-1} \mathbf{R}(\mathbf{x}_k), \tag{62}$$

where $\mathbf{J}_k$ is an approximate Jacobian of $\mathbf{R}(\mathbf{x})$. Only when $U(\mathbf{x})$ is differentiable will we have the optimality equations as $\mathbf{R}(\mathbf{x}) = \nabla U(\mathbf{x}) = \mathbf{0}$ and (62) reverts to (60).

Consider minimax as an example for the non-differentiable case. The optimality equations can be shown as (see, for instance, Hettich [78], Bandler et al. [26])

$$\mathbf{R}(\mathbf{x}, \boldsymbol{\lambda}) = \begin{bmatrix} \sum\limits_{j \in A(\mathbf{x})} \lambda_j f_j'(\mathbf{x}) \\ 1 - \sum\limits_{j \in A(\mathbf{x})} \lambda_j \\ \mathbf{z} \end{bmatrix} = \mathbf{0}, \tag{63}$$

where $\boldsymbol{\lambda}$ is a set of nonnegative Kuhn-Tucker multipliers and $A(\mathbf{x})$ is an index set, as $A(\mathbf{x}) = \{j \mid f_j(\mathbf{x}) = H_\infty(\mathbf{f})\}$, identifying the active functions. $\mathbf{z}$ has the components $f_{j_0}(\mathbf{x}) - f_j(\mathbf{x})$, $j \in A(\mathbf{x})$, $j \neq j_0$, where $j_0 \in A(\mathbf{x})$ is fixed. The approximate Jacobian $\mathbf{J}_k$ of (63) contains a mixture of the first derivatives f_j' and approximations to the Hessians f_j'' (for details see [26]). An incremental vector is obtained by solving the linear system

$$J_k \begin{bmatrix} h_k \\ \Delta\lambda_k \end{bmatrix} = -R(x_k, \lambda_k). \tag{64}$$

This example has demonstrated that the quasi-Newton method is more than just (60). It can be quite involved in cases where the ordinary gradient ∇U does not exist. A similar result for ℓ_1 can be found in [27]. Clarke [50] has introduced the concept of generalized gradient with which optimality conditions can be derived for a broad range of problems.

Quasi-Newton methods, whether in (60) or (62), all require updates of certain approximate Hessians. Many formulas have been proposed over the years. The most well-known are the Powell Symmetric Broyden (PSB) update [102], the Davidon-Fletcher-Powell (DFP) update [53,65] and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update [41,62,70,109]. They are given by, respectively,

$$\begin{aligned} B_{k+1}^{PSB} &= B_k + \frac{ws^T + sw^T}{s^T s} - \frac{w^T s\, ss^T}{(s^T s)^2}, \\ B_{k+1}^{DFP} &= B_k + \frac{wy^T + yw^T}{y^T s} - \frac{w^T s\, yy^T}{(y^T s)^2}, \\ B_{k+1}^{BFGS} &= B_k + \frac{yy^T}{y^T s} - \frac{B_k ss^T B_k}{s^T B_k s}, \end{aligned} \tag{65}$$

where $s = x_{k+1} - x_k$, $y = \nabla U(x_{k+1}) - \nabla U(x_k)$ (if B is to approximate the Hessian of U) or $y = f_j'(x_{k+1}) - f_j'(x_k)$ (if B is to approximate f_j''), and $w = y - B_k s$. Dennis and Moré [54] have given a thorough treatment of the theory underlying these updates. As they have pointed out, numerical evidence seems to support the BFGS update as the best formula for use in minimization. The interesting expression

$$B_{k+1}^{\theta} = \theta B_{k+1}^{DFP} + (1-\theta) B_{k+1}^{BFGS} \tag{66}$$

describes the Broyden family (Broyden [40], Fletcher [62]). The merits of a great many variations are often compared in terms of their preservation of positive definiteness, convergence to the true Hessian and numerical performance (see for instance Fletcher [63], Gill and Murray [69]).

When we apply (60), instead of computing $\mathbf{B}_k^{-1}$, usually a linear system is solved using LU or LDL^T factorization. Yet a more efficient approach is to update the inverse of an approximate Hessian. Suppose $\mathbf{H}_k = \mathbf{B}_k^{-1}$, then we can use

$$\mathbf{h}_k = -\alpha_k \mathbf{H}_k \nabla U(\mathbf{x}_k). \tag{67}$$

We have

$$\mathbf{H}_{k+1}^{DFP} = \mathbf{H}_k + \frac{\mathbf{s}\mathbf{s}^T}{\mathbf{s}^T\mathbf{y}} - \frac{\mathbf{H}_k \mathbf{y}\mathbf{y}^T \mathbf{H}_k}{\mathbf{y}^T \mathbf{H}_k \mathbf{y}},$$

$$\mathbf{H}_{k+1}^{BFGS} = \mathbf{H}_k + \frac{\mathbf{z}\mathbf{s}^T + \mathbf{s}\mathbf{z}^T}{\mathbf{s}^T\mathbf{y}} - \frac{\mathbf{z}^T\mathbf{y}\,\mathbf{s}\mathbf{s}^T}{(\mathbf{s}^T\mathbf{y})^2}, \tag{68}$$

where $\mathbf{z} = \mathbf{s} - \mathbf{H}_k\mathbf{y}$. The similarity between (65) and (68) is due to the fact that if $\mathbf{y} = \mathbf{B}\mathbf{s}$ then $\mathbf{s} = \mathbf{H}\mathbf{y}$. By interchanging the roles of $\mathbf{y}$ and $\mathbf{s}$, as well as $\mathbf{B}$ and $\mathbf{H}$, the DFP formula for $\mathbf{B}$ becomes the BFGS update for $\mathbf{H}$. For this reason, the BFGS formula is sometimes called the complementary DFP update.

Another important point to be considered is the line search. Ideally, α_k is chosen as the minimizer of U in the direction of line search so that $\mathbf{h}_k^T \nabla U(\mathbf{x}_k + \mathbf{h}_k) = 0$. If exact line searches are executed, Dixon [58] has shown that theoretically all members of the Broyden family (66) would have the same performance. In practice, however, exact line search is deemed too expensive and therefore replaced by other methods. An inexact line search usually limits the evaluation of U and ∇U to only a few points. Interpolation and extrapolation techniques (such as a quadratic or cubic fit) are then incorporated. The result, namely $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{h}_k$, is normally required to satisfy conditions like

$$U(\mathbf{x}_{k+1}) < U(\mathbf{x}_k),$$

$$|\mathbf{h}_k^T \nabla U(\mathbf{x}_{k+1})| \le \beta |\mathbf{h}_k^T \nabla U(\mathbf{x}_k)|, \quad \beta < 1. \tag{69}$$

Combined Methods

The distinguished advantage of a quasi-Newton method is that it enjoys fast rate of convergence near a solution. However, like the Newton method for nonlinear equations, the quasi-Newton method is not always reliable from a bad starting point.

Hald and Madsen [75,76,88] have suggested a class of 2-stage algorithms. A first-order method of the Gauss-Newton type is employed in Stage 1 to provide global convergence to a neighbourhood of a solution. When the solution is singular, Method 1 suffers from a very slow rate of convergence and a switch is made to a quasi-Newton method (Stage 2). Several switches between the two methods may take place and the switching criteria ensure the global convergence of the combined algorithm. Numerical examples of circuit applications have demonstrated a very strong performance of the approach [26,27,89,90].

Powell [103] has extended the Levenberg method and suggested a trust-region strategy which interpolates between a steepest descent step and a Newton step. When far away from the solution, the step is biased toward the steepest descent direction to make sure that it is downhill. Once close to the solution, taking a full Newton step will provide rapid final convergence.

Conjugate Gradient Methods

Some extremely large-scale engineering applications involve hundreds of variables and functions. Although the rapid advances in computer technology have enabled us to solve increasingly large problems, there may be cases in which even the storage of a Hessian matrix and the solution of an n by n linear system become unmanageable.

Conjugate gradient methods provide an alternative for such a problem. A conjugate gradient iteration is generally defined by

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \mathbf{d}_k , \\ \mathbf{d}_{k+1} &= -\nabla U(\mathbf{x}_{k+1}) + \beta_k \mathbf{d}_k , \\ \mathbf{d}_0 &= -\nabla U(\mathbf{x}_0) , \end{aligned} \tag{70}$$

where α_k is determined through a line search. Different choices of β_k lead to variations of the method. For example, the choice given by Fletcher and Reeves [66] is

$$\beta_k = [\nabla U(\mathbf{x}_{k+1})]^T \nabla U(\mathbf{x}_{k+1}) / [\nabla U(\mathbf{x}_k)]^T \nabla U(\mathbf{x}_k) \tag{71}$$

and the one due to Polak and Ribiere [98] is

$$\beta_k = [\nabla U(\mathbf{x}_{k+1}) - \nabla U(\mathbf{x}_k)]^T \nabla U(\mathbf{x}_{k+1}) / [\nabla U(\mathbf{x}_k)]^T \nabla U(\mathbf{x}_k) . \tag{72}$$

Very recently a more sophisticated formula has been proposed by Le [85] where a comprehensive comparison between various methods is also available.

A distinct advantage of conjugate gradient methods is the minimal requirement of storage. Typically three to six vectors of length n are needed, which is substantially less than the requirement by the Gauss-Newton or quasi-Newton methods. However, proper scaling or preconditioning, near-perfect line searches and appropriate restart criteria are usually necessary to ensure convergence. In general, we have to pay the price for the reduced storage by enduring a longer computation time.

Constrained Optimization

As have stated in a previous subsection, we can easily incorporate linear constraints in the form of

$$\begin{aligned} \mathbf{a}_\ell^T \mathbf{x} + b_\ell &= 0 , \quad \ell = 1, 2, ..., L_{eq} , \\ \mathbf{a}_\ell^T \mathbf{x} + b_\ell &\geq 0, \quad \ell = L_{eq} + 1, ..., L , \end{aligned} \tag{73}$$

into a linear ℓ_p problem which can then be solved using linear or quadratic programming techniques. Recall that by the Gauss-Newton method we have to solve a linearized subproblem $P(\mathbf{h},\bar{\mathbf{f}},D)$, where D normally contains only the local bounds. Clearly, we may include in D other linear constraints as well. In other words, the Gauss-Newton method can be extended to include (73).

We can also apply a quasi-Newton method to solve the optimality equations derived from the Kuhn-Tucker conditions for a constrained minimum. Due to the linearity of the constraints, the extension from an unconstrained algorithm is usually manageable. See, for example, the algorithms for linearly constrained minimax and ℓ_1 problems [26,27,75,76].

In circuit applications, it is quite common to have bounds on some variables. Easily we can express $c \geq x_i \geq d$ by a pair of linear constraints as

$$\begin{aligned} -x_i + c &\geq 0 \ , \\ x_i - d &\geq 0 \ . \end{aligned} \tag{74}$$

The positivity of model parameters is almost implied in circuit optimization. One simple trick is to let $x_i = \exp(x_i')$ and use x_i' as a variable in optimization. While x_i' is free to assume any value, $x_i > 0$ is guaranteed. As an advantageous by-product, it introduces an automatic scaling to the variables, as

$$\frac{\Delta U}{\Delta x_i'} = \frac{\Delta U}{\Delta x_i / x_i} \ , \tag{75}$$

where Δ stands for an incremental change. Many practical experiences seem to endorse this simple transformation in lieu of a more sophisticated algorithm.

The subject of treating general nonlinear constraints is much more complicated. The methods of penalty function and barrier function (see, for example, Zangwill [123], Fiacco and McCormick [61]) are relatively easy to adopt. The Han-Powell method [77, 104] is more powerful yet more complex to implement.

Actually, the ideas of penalty and barrier functions have been implied in the definition of a cost function in (33), where we use the barrier terms of a_i/ε_i to prohibit the tolerance ε_i from becoming too small and the penalty terms of $b_i t_i$ to discourage any excessive tuning. Alternatively, we may specify the minimum allowance for a component tolerance, in the form of a linear constraint on the variable ε_i: $\varepsilon_i \geq \varepsilon_{i,\min}$. Also, the tuning on ϕ_i can be

explicitly restricted by imposing $t_i \le t_{i,max}$. Such a set of linear constraints in effect specifies a maximum cost which is not to be exceeded under any circumstances.

The generalized ℓ_p function provides a perfect example of a combined penalty-barrier method. When outside the acceptable region (i.e., when $H_p(e) > 0$), the objective function is defined by (12) which is like a penalty function, and when inside it is given by (14) which is like a barrier function.

VIII. GRADIENT CALCULATION AND APPROXIMATION

A common assumption of the gradient-based ℓ_p optimization methods is that the first-order derivatives of $f_1, f_2, \ldots, f_m$ with respect to $x_1, x_2, \ldots, x_n$ are available. The gradient of the objective function, if it exists, can be readily obtained by applying the chain rule. For example, the gradient of the ℓ_p norm is given by

$$\nabla U(\mathbf{x}) = \nabla \|\mathbf{f}(\mathbf{x})\|_p = (\|\mathbf{f}\|_p)^{1-p} \sum_{j=1}^{m} (|f_j|^p / f_j)\, \mathbf{f}_j', \tag{76}$$

which exists for $p < \infty$ and when $f_j \neq 0$ for all j. If $\mathbf{f}$ is defined for multiple circuits as $\mathbf{f}(\mathbf{x}) = [\mathbf{e}^T(\boldsymbol{\phi}^0)\ \mathbf{e}^T(\boldsymbol{\phi}^1) \ldots \mathbf{e}^T(\boldsymbol{\phi}^K)]^T$, we have

$$\frac{\partial \mathbf{f}^T}{\partial \mathbf{x}} = \left[\left(\frac{\partial (\boldsymbol{\phi}^0)^T}{\partial \mathbf{x}} \frac{\partial \mathbf{e}^T(\boldsymbol{\phi}^0)}{\partial \boldsymbol{\phi}^0} \right) \cdots \left(\frac{\partial (\boldsymbol{\phi}^K)^T}{\partial \mathbf{x}} \frac{\partial \mathbf{e}^T(\boldsymbol{\phi}^K)}{\partial \boldsymbol{\phi}^K} \right) \right]. \tag{77}$$

Furthermore, since each error function $e_j(\boldsymbol{\phi}^k)$ is derived from a circuit response $F_j(\boldsymbol{\phi}^k)$ and a constant specification (see Section II), we need to consider only the sensitivities of $F_j(\boldsymbol{\phi}^k)$. The application of the chain rule can be illustrated by

$$\frac{\partial F_j}{\partial \boldsymbol{\phi}^k} \rightarrow \frac{\partial \mathbf{e}^T}{\partial \boldsymbol{\phi}^k} \rightarrow \frac{\partial \mathbf{f}^T}{\partial \mathbf{x}} \rightarrow \nabla U(\mathbf{x}). \tag{78}$$

In the following, we first consider linear circuits in the frequency domain, in which case exact sensitivity analysis is usually possible. We also point out some common obstacles to the implementation of sensitivity calculations in practice. Then some algorithms of gradient approximation are discussed.

Sensitivities for a Nodal Description

Assume that a nodal description of the circuit is available. For simplicity, we further assume an admittance matrix. The formulas are, of course, applicable to an impedance or hybrid matrix.

We have

$$\mathbf{Y}\,\mathbf{V} = \mathbf{I} \tag{79}$$

where $\mathbf{Y}$ is the N by N admittance matrix, $\mathbf{V}$ the nodal voltages and $\mathbf{I}$ the excitation vector. Differentiating (79) w.r.t. a generic variable ϕ gives

$$\frac{\partial \mathbf{V}}{\partial \phi} = -\,\mathbf{Y}^{-1}\,\frac{\partial \mathbf{Y}}{\partial \phi}\,\mathbf{V}. \tag{80}$$

To select the sensitivity for a particular voltage of interest, say, V_ℓ, we define a unit vector $\mathbf{u}_\ell$ which is the ℓth column vector of an N by N identity matrix (its ℓth element is 1 and the others are zeros). Premultiply (80) by $\mathbf{u}^T_\ell$,

$$\frac{\partial V_\ell}{\partial \phi} = -\,\mathbf{u}_\ell^T\,\mathbf{Y}^{-1}\,\frac{\partial \mathbf{Y}}{\partial \phi}\,\mathbf{V} = -\,\hat{\mathbf{V}}^T\,\frac{\partial \mathbf{Y}}{\partial \phi}\,\mathbf{V} = -\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\partial Y_{ij}}{\partial \phi}\,\hat{V}_i\,V_j, \tag{81}$$

where we define an adjoint system by

$$\mathbf{Y}^T\,\hat{\mathbf{V}} = \mathbf{u}_\ell. \tag{82}$$

For example, consider a capacitor connected between nodes a and b. The parameter C appears in four places in $\mathbf{Y}$: as $j\omega C$ in Y_{aa} and Y_{bb} and as $-j\omega C$ in Y_{ab} and Y_{ba}. Therefore,

$$\frac{\partial V_\ell}{\partial C} = -j\omega\,(\hat{V}_a V_a + \hat{V}_b V_b - \hat{V}_a V_b - \hat{V}_b V_a) = -j\omega\,(\hat{V}_a - \hat{V}_b)(V_a - V_b). \tag{83}$$

If we solve the original circuit (79) by LU factorization, then the adjoint solution (82) requires minimal extra effort.

It is also possible to arrive at the same results from Tellegen's theorem (see, for instance, Director and Rohrer [56] and Bandler [7]). The concise derivation through matrix algebra as shown here was first presented by Branin [37].

Sensitivities for Terminated Two-ports

If the nodal equations (79) describe an unterminated network and we focus our attention on a pair of input and output variables, then the circuit is characterized by an unterminated two-port

$$\begin{bmatrix} V_1 \\ V_N \end{bmatrix} = \begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{bmatrix} \begin{bmatrix} I_1 \\ I_N \end{bmatrix}, \tag{84}$$

where node 1 being the input and node N the output is just a convenient assumption. Writing (84) as $\mathbf{V}_p = \mathbf{z}\,\mathbf{I}_p$, it can be shown that the two-port open-circuit impedance matrix is given by

$$\mathbf{z} = \begin{bmatrix} \mathbf{u}_1^T \mathbf{Y}^{-1} \mathbf{u}_1 & \mathbf{u}_1^T \mathbf{Y}^{-1} \mathbf{u}_N \\ \mathbf{u}_N^T \mathbf{Y}^{-1} \mathbf{u}_1 & \mathbf{u}_N^T \mathbf{Y}^{-1} \mathbf{u}_N \end{bmatrix} = \begin{bmatrix} p_1 & q_1 \\ p_N & q_N \end{bmatrix}, \tag{85}$$

where $\mathbf{p}$ and $\mathbf{q}$ are solutions of, respectively,

$$\begin{aligned} \mathbf{Y}\,\mathbf{p} &= \mathbf{u}_1, \\ \mathbf{Y}\,\mathbf{q} &= \mathbf{u}_N. \end{aligned} \tag{86}$$

Following (81), we obtain these sensitivity expressions:

$$\frac{\partial \mathbf{z}}{\partial \phi} = \frac{1}{\partial \phi} \begin{bmatrix} \partial p_1 & \partial q_1 \\ \partial p_N & \partial q_N \end{bmatrix} = - \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{\partial Y_{ij}}{\partial \phi} \begin{bmatrix} \hat{p}_i p_j & \hat{p}_i q_j \\ \hat{q}_i p_j & \hat{q}_i q_j \end{bmatrix}, \tag{87}$$

where two adjoint systems are defined by

$$\begin{aligned} \mathbf{Y}^T \hat{\mathbf{p}} &= \mathbf{u}_1, \\ \mathbf{Y}^T \hat{\mathbf{q}} &= \mathbf{u}_N. \end{aligned} \tag{88}$$

Assume that the two-port is terminated by a load Y_L and a source $J = 1A$ with an admittance Y_S. Then

$$\begin{bmatrix} I_1 \\ I_N \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} Y_S & 0 \\ 0 & Y_L \end{bmatrix} \begin{bmatrix} V_1 \\ V_N \end{bmatrix}. \tag{89}$$

Denoting

$$\mathbf{T} = \begin{bmatrix} Y_S & 0 \\ 0 & Y_L \end{bmatrix}, \tag{90}$$

we can express (89) more concisely as $\mathbf{I}_p = \mathbf{u}_1 - \mathbf{T}\,\mathbf{V}_p$. From (84) and (89), we can solve for $\mathbf{V}_p$ as

$$\mathbf{V}_p = (1 + \mathbf{z}\mathbf{T})^{-1} \mathbf{z}\mathbf{u}_1 \tag{91}$$

and, after some algebraic manipulations, obtain its sensitivity expression as

$$\frac{\partial \mathbf{V}_p}{\partial \phi} = (1 + zT)^{-1}\left(\frac{\partial z}{\partial \phi}\mathbf{I}_p - z\frac{\partial \mathbf{T}}{\partial \phi}\mathbf{V}_p\right). \tag{92}$$

We notice that in order to complete the derivation, four linear systems have been defined. If the network is reciprocal then $\mathbf{Y}^T = \mathbf{Y}$ and consequently (88) is identical to (86), which means that only two systems need to be solved. Furthermore, it has been shown that for lossless two-ports one circuit analysis will suffice (Orchard et al. [94], also Bandler et al. [14]). In some cases second-order sensitivities can be derived as well [15]. It is not difficult to see that the above results can be extended for general multi-ports, the difference being that more linear systems will have to be defined and solved. We should emphasize the fact that in any of these cases, only one LU factorization of $\mathbf{Y}$ is required.

Sensitivities for Frequency Responses

In the microwave area, the use of reflection coefficient, return loss, insertion loss and scattering parameters is very popular. Once we have the sensitivities of the two-port voltages and currents, to derive the corresponding formulas for those frequency responses is more of an algebraic exercise. We simply summarize some of the basic results in Table IV.

Obstacles to Practical Implementation

We ought to recognize that an explicit and elegant sensitivity expression is not always available. For time-domain responses and nonlinear circuits an exact formula may not exist. Even for linear circuits in the frequency domain, large-scale networks present new problems which need to be addressed.

Often, a large-scale network can be described through compounded and interconnected subnetworks. Many commercial CAD packages such as SUPER-COMPACT [113] and TOUCHSTONE [118] have facilitated such a block structure. In this case, one possible approach would be to assemble the overall nodal matrix and solve the system of

equations using sparse techniques (see, e.g., Duff [59], Gustavson [71], Hachtel et al. [72]). Another possibility is to rearrange the overall nodal matrix into a bordered block structure which is then solved using the Sherman-Morrison-Woodbury formula [73,110]. Sometimes it is also possible to develop efficient formulas for a special structure, such as the approach of Bandler et al. [22] for branched cascaded networks.

In conjunction with these approaches, the analyses and sensitivity calculations as outlined in the previous subsections can be carried out at the subnetwork level. Perhaps the most perplexing and time-consuming part of the task is to devise an index scheme through which pieces of lower-level information can be brought into the overall sensitivity expression. It may also require a large amount of memory storage for the various intermediate results. Partly due to these difficulties, methods of exact sensitivity calculations have yet to find their way into general-purpose CAD software packages, although the concept of adjoint network has been in existence for nearly two decades and has had success in many specialized applications.

Gradient Approximations

In the case where either exact sensitivities do not exist or they are too difficult to calculate, we can implement gradient approximations in order to furnish the necessary link between a simulation module which provides function values and an optimization module which demands gradients.

Approximations to second-order derivatives have been extensively studied in the context of quasi-Newton methods. Those results are not directly applicable to gradient approximations because certain important properties of a Hessian such as symmetry and positive definiteness are not generally relevant to a Jacobian.

Traditionally, approximate first-order derivatives are obtained by perturbations (finite differences) as

$$\frac{\partial f_j(\mathbf{x})}{\partial x_i} \simeq \frac{f_j(\mathbf{x} + h\mathbf{u}_i) - f_j(\mathbf{x})}{h} . \tag{93}$$

It gives a reliable result but the computational labor involved grows in proportion to the dimension of the problem.

Recently, Bandler et al. [19-21] have proposed an effective and flexible approach to gradient approximations. Their work has extended and improved upon the earlier results by Madsen [87] and Zuberek [124]. It is a hybrid approach combining perturbations, the Broyden update [39] and the special iterations of Powell [101]. Let $(\mathbf{g}_j)_k$ be an approximation to $\mathbf{f}_j'$. The Broyden rank-one formula is given by

$$(\mathbf{g}_j)_{k+1} = (\mathbf{g}_j)_k + \frac{f_j(\mathbf{x}_k + \mathbf{h}_k) - f_j(\mathbf{x}_k) - (\mathbf{g}_j)_k^T \mathbf{h}_k}{\mathbf{h}_k^T \mathbf{h}_k} \mathbf{h}_k . \tag{94}$$

If $\mathbf{x}_k$ and $\mathbf{x}_k + \mathbf{h}_k$ are iterates of optimization then (94) does not require extra function evaluations. Powell's special iterations are incorporated to overcome a particular deficiency of the Broyden update by guaranteeing strictly linearly independent directions. A weighted update has also been suggested which is especially suitable for a multi-circuit structure. Such an algorithm has been integrated with minimax and ℓ_1 optimization for circuit applications. It has proved to be efficient and reliable through practical examples in areas including multiplexer design, FET modeling and worst-case design.

VII. CONCLUSIONS

In this review, we have formulated realistic circuit design and modeling problems and described their solution methods. Models, variables and functions at different levels, as well as the associated tolerances and uncertainties have been identified. The concepts of design centering, tolerancing and tuning have been discussed. Recent advances in statistical design, yield enchancement and robust modeling techniques suitable for microwave CAD have been exposed in detail. State-of-the-art optimization techniques have been addressed from both

the theoretical and algorithmic points of view. Sensitivity calculations and gradient approximations have been discussed in some detail and useful formulas have been given.

We have concentrated on aspects that are felt to be immediately relevant to and necessary for modern microwave CAD. There are, of course, other related subjects that have not or not adequately been treated in this paper. Notably among these are special techniques for very large systems (Geoffrion [67, 68], Haimes [74], Lasdon [82]), third generation simulation techniques (Hachtel and Sangiovanni-Vincentelli [73]), fault diagnosis (Bandler and Salama [33]), supercomputer-aided CAD (Rizzoli et al. [106]), and the new automated decomposition approach to large scale optimization (Bandler and Zhang [34]).

This paper is particularly timely as software, based on techniques which we have described, is being integrated by EEsof Inc. into TOUCHSTONE and Compact Software Inc. into SUPER-COMPACT.

ACKNOWLEDGEMENTS

The authors would like to thank Dr. K.C. Gupta, Guest Editor of this Special Issue on Computer Aided Design for his invitation to write this review paper. The authors must acknowledge original work done by several researchers which has been integrated into our presentation, including Drs. H.L. Abdel-Malek, R.M. Biernacki, C. Charalambous, S. Daijavad, W. Kellermann, P.C. Liu, K. Madsen, M.R.M. Rizk, H. Tromp and Q.J. Zhang. M.L. Renault is thanked for her contributions, including assistance in preparing data, programs and results. The opportunity EEsof Inc. provided to develop state-of-the-art optimizers as practical design tools, through interaction with Drs. W.H. Childs, C.H. Holmes and D. Morton, is appreciated. Cooperation with ComDev Ltd. in preparing relevant filter data is acknowledged. Finally, thanks are extended to Dr. R.A. Pucel of Raytheon Company, Research Division, Lexington, MA, for reviving the first author's interest in design centering and yield optimization and his sustained initiatives in advocating advanced mathematical techniques to CAD of microwave integrated circuits.

REFERENCES

[1] H.L. Abdel-Malek and J.W. Bandler, "Yield optimization for arbitrary statistical distributions, part I: theory", IEEE Trans. Circuits and Systems, vol. CAS-27, 1980, pp. 245-253.

[2] H.L. Abdel-Malek and J.W. Bandler, "Yield optimization for arbitrary statistical distributions, part II: implementation", IEEE Trans. Circuits and Systems, vol. CAS-27, 1980, pp. 253-262.

[3] D. Agnew, "Improved minimax optimization for circuit design", IEEE Trans. Circuits and Systems, vol. CAS-28, 1981, pp. 791-803.

[4] A.E. Atia, "Computer-aided design of waveguide multiplexers", IEEE Trans. Microwave Theory Tech., vol. MTT-22, 1974, pp. 332-336.

[5] A.E. Atia and A.E. Williams, "New types of waveguide bandpass filters for satellite transponders", COMSAT Technical Review, vol. 1, 1971, pp. 21-43.

[6] J.W. Bandler, "Optimization methods for computer-aided design", IEEE Trans. Microwave Theory and Tech., vol. MTT-17, 1969, pp. 533-552.

[7] J.W. Bandler, "Computer-aided circuit optimization", in Modern Filter Theory and Design, G.C. Temes and S.K. Mitra, eds., John Wiley, New York, 1973, pp. 211-271.

[8] J.W. Bandler, "Engineering modelling and design subject to model uncertainties and manufacturing tolerances", in Methodology in Systems Modelling and Simulation, B.P. Zeigler, et al., Eds., North Holland, Amsterdam, 1979, pp. 399-421.

[9] 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.

[10] J.W. Bandler, R.M. Biernacki and A.E. Salama, "A linear programming approach to fault location in analog circuits", Proc. IEEE Int. Symp. Circuits and Systems (Chicago, IL), 1981, pp. 256-260.

[11] J.W. Bandler, R.M. Biernacki, A.E. Salama and J.A. Starzyk, "Fault isolation in linear analog circuits using the L_1 norm", Proc. IEEE Int. Symp. Circuits and Systems (Rome, Italy), 1982, pp. 1140-1143.

[12] J.W. Bandler and C. Charalambous, "Theory of generalized least pth approximation", IEEE Trans. Circuit Theory, vol. CT-19, 1972, pp. 287-289.

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

[14] J.W. Bandler, S.H. Chen and S. Daijavad, "Simple derivation of a general sensitivity formula for lossless two-ports", Proc. IEEE, vol. 73, 1985, pp. 165-166.

[15] J.W. Bandler, S.H. Chen and S. Daijavad, "Exact sensitivity analysis for optimization of multi-coupled cavity filters", Int. J. Circuit Theory and Applic., vol. 14, 1986, pp. 63-77.

[16] J.W. Bandler, S.H. Chen and S.Daijavad, "Microwave device modeling using efficient ℓ_1 optimization: a novel approach", IEEE Trans. Microwave Theory Tech., vol. MTT-34, 1986, pp. 1282-1293.

[17] J.W. Bandler, S.H. Chen, S. Daijavad and W. Kellermann, "Optimal design of multi-cavity filters and contiguous-band multiplexers", Proc. 14th European Microwave Conf.(Liege, Belgium), pp. 863-868.

[18] J.W. Bandler, S.H. Chen, S. Daijavad, W. Kellermann, M. Renault and Q.J. Zhang, "Large scale minimax optimization of microwave multiplexers", Proc. 16th European Microwave Conf. (Dublin, Ireland), 1986, pp. 435-440.

[19] J.W. Bandler, S.H. Chen, S. Daijavad and K. Madsen, "Efficient gradient approximations for nonlinear optimization of circuits and systems", Proc. IEEE Int. Symp. Circuits and Systems (San Jose, CA), 1986, pp. 964-967.

[20] J.W. Bandler, S.H. Chen, S. Daijavad and K. Madsen, "Efficient optimization with integrated gradient approximations, Part I: algorithms", submitted to IEEE Trans. Microwave Theory Tech. for publication, 1987.

[21] J.W. Bandler, S.H. Chen, S. Daijavad and K. Madsen, "Efficient optimization with integrated gradient approximations, Part II: implementation", submitted to IEEE Trans. Microwave Theory Tech. for publication, 1987.

[22] J.W. Bandler, S. Daijavad and Q.J. Zhang, "Computer aided design of branched cascaded networks", Proc. IEEE Int. Symp. Circuits and Systems (Kyoto, Japan), 1985, pp. 1579-1582.

[23] J.W. Bandler, S. Daijavad and Q.J. Zhang, "Exact simulation and sensitivity analysis of multiplexing networks", IEEE Trans. Microwave Theory Tech., vol. MTT-34, 1986, pp. 93-102.

[24] J.W. Bandler, M.A. El-Kady, W. Kellermann and W.M. Zuberek, "An optimization approach to the best alignment of manufactured and operating systems", Proc. IEEE Int. Symp. Circuits and Systems (Newport Beach, CA), 1983, pp. 542-545.

[25] J.W. Bandler and W. Kellermann, "Selected topics in optimal design centering, tolerancing and tuning", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-83-28, 1983.

[26] J.W. Bandler, W. Kellermann and K. Madsen, "A superlinearly convergent minimax algorithm for microwave circuit design", IEEE Trans. Microwave Theory Tech., vol. MTT-33, 1985, pp. 1519-1530.

[27] J.W. Bandler, W. Kellermann and K. Madsen, "A nonlinear ℓ_1 optimization algorithm for design, modelling and diagnosis of networks", IEEE Trans. Circuits and Systems, vol. CAS-34, 1987, pp.174-181.

[28] J.W. Bandler, P.C. Liu and J.H.K. Chen, "Worst case network tolerance optimization", IEEE Trans. Microwave Theory Tech., vol. MTT-23, 1975, pp. 630-641.

[29] 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, 1976, pp. 155-165.

[30] J.W. Bandler, P.C. Liu and H. Tromp, "Integrated approach to microwave design", IEEE Trans. Microwave Theory Tech., vol. MTT-24, 1976, pp. 584-591.

[31] J.W. Bandler and M.R.M. Rizk, "Optimization of electrical circuits", Math. Programming Study, vol. 11, 1979, pp.1-64.

[32] J.W. Bandler and A.E. Salama, "Functional approach to microwave postproduction tuning", IEEE Trans. Microwave Theory Tech., vol. MTT-33, 1985, pp. 302-310.

[33] J.W. Bandler and A.E. Salama, "Fault diagnosis of analog circuits", Proc. IEEE, vol. 73, 1985, pp. 1279-1325.

[34] J.W. Bandler and Q.J. Zhang, "An automatic decomposition technique for device modelling and large circuit design", IEEE Int. Microwave Symp. Digest (Las Vegas, NA), 1987.

[35] R.H. Bartels and A.R. Conn, "An approach to nonlinear ℓ_1 data fitting", Computer Science Department, University of Waterloo, Waterloo, Canada, Report CS-81-17, 1981.

[36] R.M. Biernacki and M.A. Styblinski, "Statistical circuit design with a dynamic constraint approximation scheme", Proc. IEEE Int. Symp. Circuits and Systems (San Jose, CA), 1986, pp. 976-979.

[37] F.H. Branin, Jr., "Network sensitivity and noise analysis simplified", IEEE Trans. Circuit Theory, vol. CT-20, 1973, pp. 285-288.

[38] R.K. Brayton, G.D. Hachtel and A.L. Sangiovanni-Vincentelli, "A survey of optimization techniques for integrated-circuit design", Proc. IEEE, vol. 69, 1981, pp. 1334-1362.

[39] C.G. Broyden, "A class of methods for solving nonlinear simultaneous equations", Math. Comp., vol. 19, 1965, pp. 577-593.

[40] C.G. Broyden, "Quasi-Newton methods and their application to function minimization", Math. Comp., vol. 21, 1967, pp. 368-381.

[41] C.G. Broyden, "A new double-rank minimization algorithm", Notices American Math. Society, vol. 16, 1969, p. 670.

[42] E.M. Butler, "Realistic design using large-change sensitivities and performance contours", IEEE Trans. Circuit Theory, vol. CT-18, 1971, pp. 58-66.

[43] D.A. Calahan, Computer-Aided Network Design (Revised Edition), McGraw Hill, New York, 1972.

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

[45] C. Charalambous, "Minimax design of recursive digital filters", Computer Aided Design, vol. 6, 1974, pp. 73-81.

[46] C. Charalambous, "Nonlinear least pth optimization and nonlinear programming", Math. Programming, vol. 12, 1977, pp. 195-225.

[47] C. Charalambous and A.R. Conn, "Optimization of microwave networks", IEEE Trans. Microwave Theory Tech., vol. MTT-23, 1975, pp. 834-838.

[48] M.H. Chen, "Current state-of-the-art technology on contiguous-band multiplexer", Proc. IEEE Int. Symp. Circuits and Systems (Kyoto, Japan), 1985, pp. 1583-1586.

[49] M.H. Chen, F. Assal and C.Mahle, "A contiguous band multiplexer", COMSAT Technical Review, vol. 6, 1976, pp. 285-306.

[50] F.H. Clarke, "Generalized gradients and applications", Trans. American Math. Society, vol. 205, 1975, pp. 247-262.

[51] ComDev Ltd., 155 Sheldon Drive, Cambridge, Ontario, Canada N1R 7H6, private communications, 1986.

[52] S. Daijavad, "Design and modelling of microwave circuits using optimization methods", Ph.D. Thesis, McMaster University, Hamilton, Canada, 1986.

[53] W.C. Davidon, "Variable metric method for minimization", Rep. ANL-5990 Rev., Argonne National Laboratories, Argonne, IL, 1959.

[54] J.E. Dennis, Jr. and J.J. Moré, "Quasi-Newton methods, motivation and theory", SIAM Review, vol. 19, 1977, pp. 46-89.

[55] S.W. Director and G.D. Hachtel, "The simplicial approximation approach to design centering", IEEE Trans. Circuits and Systems, vol. CAS-24, 1977, pp. 363-372.

[56] S.W. Director and R.A. Rohrer, "Generalized adjoint network and network sensitivities", IEEE Trans. Circuit Theory, vol. CT-16, 1969, pp. 318-323.

[57] S.W. Director and R.A. Rohrer, "Automated network design: The frequency domain case", IEEE Trans. Circuit Theory, vol. CT-16, 1969, pp. 330-337.

[58] L.C.W. Dixon, "Quasi-Newton algorithms generate identical points", Math. Programming, vol. 2, 1972, pp. 383-387.

[59] I.S. Duff, "A survey of sparse matrix research", Proc. IEEE, vol. 65, 1977, pp. 500-535.

[60] N.J. Elias, "New statistical methods for assigning device tolerances", Proc. IEEE Int. Symp. Circuits and Systems (Newton, MA), 1975, pp. 329-332.

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

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

[63] R. Fletcher, "A survey of algorithms for unconstrained optimization", in Numerical Methods for Unconstrained Optimization, W. Murray, ed., Academic Press, London, 1972.

[64] R. Fletcher, Practical Methods of Optimization: Volume 2, Constrained Optimization, John Wiley, Chichester, England, 1981.

[65] R. Fletcher and M.J.D. Powell, "A rapidly convergent descent method for minimization", Comput. J., vol. 6, 1963, pp. 163-168.

[66] R. Fletcher and C.M. Reeves, "Function minimisation by conjugate gradients", Comput. J., vol. 7, 1964, pp.149-154.

[67] A.M. Geoffrion, "Elements of large-scale mathematical programming part I: concepts", Management Science, vol. 16, 1970, pp.652-675.

[68] A.M. Geoffrion, "Elements of large-scale mathematical programming part II: synthesis of algorithms and bibliography", Management Science, vol. 16, 1970, pp.676-691.

[69] P.E. Gill and W. Murray, "Quasi-Newton methods for unconstrained minimization", J. Inst. Math. Appl., vol. 9, 1972, pp. 91-108.

[70] D. Goldfarb, "A family of variable-metric methods derived by variational means", Math. Comp., vol. 24, 1970, pp. 23-26.

[71] F.G. Gustavson, "Some basic techniques for solving sparse systems of linear equations", in Sparse Matrices and Their Applications, D.J. Rose and R.A. Willoughby, Eds., Plenum Press, New York, 1971.

[72] G.D. Hachtel, R.K. Brayton and F.G. Gustavson, "The sparse tableau approach to network analysis and design", IEEE Trans. Circuit Theory, vol. CT-18, 1971, pp. 101-113.

[73] G.D. Hachtel and A.L. Sangiovanni-Vincentelli, "A survey of third-generation simulation techniques", Proc. IEEE, vol. 69, 1981, pp. 1264-1280.

[74] Y.Y. Haimes, Ed., Large Scale System, North Holland, Amsterdam, 1982.

[75] J. Hald and K. Madsen, "Combined LP and quasi-Newton methods for minimax optimization", Math. Programming, vol. 20, 1981, pp. 49-62.

[76] J. Hald and K. Madsen, "Combined LP and quasi-Newton methods for nonlinear ℓ_1 optimization", SIAM J. Numerical Analysis, vol. 22, 1985, pp. 68-80.

[77] S.P. Han, "A globally convergent method for nonlinear programming", J. Opt. Theory Applic., vol. 22, 1977, pp. 297-309.

[78] R. Hettich, "A Newton-method for nonlinear Chebyshev approximation", in Approximation Theory, R. Schaback and K. Scherer, Eds., Lecture Notes in Mathematics, 556 (Springer, Berlin, 1976), pp. 222-236.

[79] B.J. Karafin, "The optimum assignment of component tolerances for electrical networks", The Bell System Technical Journal, vol. 50, 1971, pp. 1225-1242.

[80] H.W. Kuhn and A.W. Tucker, "Non-linear programming", Proc. 2nd Symp. on Math. Statistics and Probability, Berkeley, CA, University of California Press, 1951, pp. 481-493.

[81] K.R. Laker, M.S. Ghausi and J.J. Kelly, "Minimum sensitivity active (leapfrog) and passive ladder bandpass filters", IEEE Trans. Circuits and Systems, vol. CAS-22, 1975, pp. 670-677.

[82] L.S. Lasdon, Optimization Theory for Large Systems, MacMillan, New York, 1970.

[83] L.S. Lasdon, D.F. Suchman and A.D. Waren, "Nonlinear programming applied to linear array design", J. Acoust. Soc. Am., vol. 40, 1966, pp. 1197-1200.

[84] L.S. Lasdon and A.D. Waren, "Optimal design of filters with bounded, lossy elements", IEEE Trans. Circuit Theory, vol. CT-13, 1966, pp. 175-187.

[85] D. Le, "A fast and robust unconstrained optimization method requiring minimum storage", Math. Programming, vol. 32, 1985, pp. 41-68.

[86] K. Levenberg, "A method for the solution of certain problems in least squares", Quart. Appl. Math., vol. 2, 1944, pp. 164-168.

[87] K. Madsen, "Minimax solution of nonlinear equations without calculating derivatives", Math. Programming Study, vol. 3, 1975, pp. 110-126.

[88] K. Madsen, "Minimization of non-linear approximation functions", Dr. techn. thesis, Institute of Numerical Analysis, Tech. Univ. of Denmark, DK2800 Lyngby, Denmark, 1985.

[89] K. Madsen and H. Schjaer-Jacobsen, "New algorithms for worst case tolerance optimization", Proc. IEEE Int. Symp. Circuits and Systems (New York), 1978, pp. 681-685.

[90] K. Madsen, H. Schjaer-Jacobsen and J. Voldby, "Automated minimax design of networks", IEEE Trans. Circuits and Systems, vol. CAS-22, 1975, pp. 791-796.

[91] D. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters", SIAM J. Appl. Math., vol. 11, 1963, pp. 431-441.

[92] J.J. Moré, "Recent developments in algorithms and software for trust region methods", in Mathematical Programming, The State of the Art (Bonn 1982), Springer Verlag, pp. 258-287.

[93] D.D. Morrison, "Optimization by least squares", SIAM J. Numerical Analysis, vol. 5, 1968, pp. 83-88.

[94] H.J. Orchard, G.C. Temes and T. Cataltepe, "Sensitivity formulas for terminated lossless two-ports", IEEE Trans. Circuits and Systems, vol. CAS-32, 1985, pp. 459-466.

[95] M.R. Osborne and G.A. Watson, "An algorithm for minimax optimization in the nonlinear case", Comput. J., vol. 12, 1969, pp.63-68.

[96] M.R. Osborne and G.A. Watson, "On an algorithm for discrete nonlinear ℓ_1 approximation", Comput. J., vol. 14, 1971, pp. 184-188.

[97] J.F. Pinel and K.A. Roberts, "Tolerance assignment in linear networks using nonlinear programming", IEEE Trans. Circuit Theory, vol. CT-19, 1972, pp. 475-479.

[98] E. Polak, Computational Methods in Optimization: A Unified Approach, Academic Press, New York, 1971, pp. 53-54.

[99] E. Polak, "An implementable algorithm for the optimal design centering, tolerancing, and tuning problem", J. Opt. Theory Applic., vol. 37, 1982, pp. 45-67.

[100] E. Polak and A. Sangiovanni-Vincentelli, "Theoretical and computational aspects of the optimal design centering, tolerancing, and tuning problem", IEEE Trans. Circuits and Systems, vol. CAS-26, 1979, pp. 795-813.

[101] M.J.D. Powell, "A Fortran subroutine for unconstrained minimization, requiring first derivatives of the objective functions", Atomic Energy Research Establishment, Harwell, Berkshire, England, Report AERE-R.6469, 1970, pp. 20-27.

[102] M.J.D. Powell, "A new algorithm for unconstrained optimization", in Nonlinear Programming, J.B. Rosen, O.L. Mangasarian and K. Ritter, eds., Academic Press, New York, 1970.

[103] M.J.D. Powell, "A hybrid method for nonlinear equations", in Numerical Methods for Nonlinear Algebraic Equations, P. Rabinowitz, ed., Gordon and Breach, London, 1970.

[104] M.J.D. Powell, "Algorithms for nonlinear constraints that use Lagrangian functions", Math. Programming, vol. 14, 1978, pp. 224-248.

[105] R.A. Pucel, Raytheon Company, Research Division, Lexington, MA 02173, FET data, private communications, 1986.

[106] V. Rizzoli, M. Ferlito and A. Neri, "Vectorized program architectures for supercomputer-aided circuit design", IEEE Trans. Microwave Theory Tech., vol. MTT-34, 1986, pp. 135-141.

[107] L.E. Scales, Introduction to Non-linear Optimization, Springer-Verlag New York Inc., New York, 1985.

[108] J. Schoeffler, "The synthesis of minimum sensitivity networks", IEEE Trans. Circuit Theory, vol. CT-11, 1964, pp. 271-276.

[109] D.F. Shanno, "Conditioning of quasi-Newton methods for function minimization", Math. Comp., vol. 24, 1970, pp. 647-656.

[110] J.Sherman and W.J. Morrison, "Adjustment of an inverse matrix corresponding to changes in the elements of a given column or row of the original matrix", Annu. Math. Statist., vol. 20, 1949, p. 621.

[111] K. Singhal and J.F. Pinel, "Statistical design centering and tolerancing using parametric sampling", IEEE Trans. Circuits and Systems, vol. CAS-28, 1981, pp. 692-701.

[112] R.S. Soin and R. Spence, "Statistical exploration approach to design centering", IEE. Proc., vol.127, Pt. G., 1980, pp.260-269.

[113] SUPER-COMPACT User's Manual, Communications Consulting Corp., Patterson, NJ 07504, May 1986.

[114] K.S. Tahim and R. Spence, "A radial exploration approach to manufacturing yield estimation and design centering", IEEE Trans. Circuits and Systems, vol. CAS-26, 1979, pp. 768-774.

[115] T.S. Tang and M.A. Styblinski, "Yield gradient estimation for non-differentiable density functions using convolution techniques and their application to yield optimization", Proc. IEEE Int. Symp. Circuits and Systems (San Jose, CA), 1986, pp. 1306-1309.

[116] G.C. Temes and D.A. Calahan, "Computer-aided network optimization the state-of-the-art", Proc. IEEE, vol. 55, 1967, pp. 1832-1863.

[117] G.C. Temes and D.Y.F. Zai, "Least pth approximation", IEEE Trans. Circuit Theory, vol. CT-16, 1969, pp. 235-237.

[118] TOUCHSTONE User's Manual, EEsof Inc., Westlake Village, CA 91362, Aug. 1985.

[119] H. Tromp, "The generalized tolerance problem and worst case search", Conf. Computer-aided Design of Electronic and Microwave Circuits and Systems (Hull, England), 1977, pp. 72-77.

[120] H. Tromp, "Generalized worst case design, with applications to microwave networks", Doctoral Thesis (in Dutch), Faculty of Engineering, University of Ghent, Ghent, Belgium, 1978.

[121] A.D. Waren, L.S. Lasdon and D.F. Suchman, "Optimization in engineering design", Proc. IEEE, vol. 55, 1967, pp. 1885-1897.

[122] E. Wehrhahn and R. Spence, "The performance of some design centering methods", Proc. IEEE Int. Symp. Circuits and Systems (Montreal, Canada), 1984, pp. 1424-1438.

[123] W.I. Zangwill, "Nonlinear programming via penalty functions", Management Science, vol. 13, 1967, pp 344-358.

[124] W.M. Zuberek, "Numerical approximation of gradients for circuit optimization", Proc. 27th Midwest Symp. Circuits and Systems (Morgantown, WV), 1984, pp. 200-203.

TABLE I

STATISTICAL DESIGN OF A LOW-PASS FILTER USING

PARAMETRIC SAMPLING TECHNIQUE [111]

Component	Starting Point	Worst-Case Design		Minimum Cost Design	
ϕ_i	ϕ_i^0	ϕ_i^0	$\varepsilon_i/\phi_i^0(\%)$	ϕ_i^0	$\varepsilon_i/\phi_i^0(\%)$
x_1	0.2251	0.21200	.62	0.21069	2.90
x_2	0.2494	0.23300	.58	0.23299	3.71
x_3	0.2523	0.23499	.45	0.23519	2.00
x_4	0.2494	0.23300	.58	0.23169	3.40
x_5	0.2251	0.21200	.62	0.21430	2.76
x_6	0.2149	0.23499	.85	0.23340	4.72
x_7	0.3636	0.38902	.57	0.39240	2.63
x_8	0.3761	0.40201	.51	0.39840	2.50
x_9	0.3761	0.40201	.51	0.40589	2.36
x_{10}	0.3636	0.38902	.57	0.38542	2.54
x_{11}	0.2149	0.23499	.85	0.23020	2.83
Yield		100%		78.67%	
Cost		18.68		5.00	

Independent normal distributions are assumed for each component with mean values ϕ_i^0. The tolerances ε_i are related to the standard deviations σ_i by $\varepsilon_i = 3\sigma_i$. The yield is estimated based on 300 samples.

TABLE II

STATISTICAL DESIGN OF A LOW-PASS FILTER USING

GENERALIZED ℓ_1 CENTERING TECHNIQUE

Component ϕ_i	Nominal Design $\phi_i^{0,0}$	Case 1 $\phi_i^{0,1}$	Case 2 $\phi_i^{0,2}$	Case 3 $\phi_i^{0,3}$
x_1	0.2251	0.21954	0.21705	0.21530
x_2	0.2494	0.25157	0.24677	0.23838
x_3	0.2523	0.25529	0.24784	0.24120
x_4	0.2494	0.24807	0.24019	0.23687
x_5	0.2251	0.22042	0.21753	0.21335
x_6	0.2149	0.22627	0.23565	0.23093
x_7	0.3636	0.36739	0.37212	0.38225
x_8	0.3761	0.36929	0.38012	0.39023
x_9	0.3761	0.37341	0.38371	0.39378
x_{10}	0.3636	0.36732	0.37716	0.38248
x_{11}	0.2149	0.22575	0.22127	0.23129
Yield	49%	77.67%	79.67%	83.67%
Number of samples used for design		50	100	100
Starting point		$\Phi^{0,0}$	$\Phi^{0,1}$	$\Phi^{0,2}$
Number of iterations		16	18	13
CPU time (VAX 8600)		10 sec.	30 sec.	26 sec.

Independent uniform distributions are assumed for each component with fixed tolerances $\varepsilon_i = 1.5\%\ \phi_i^0$. The yield is estimated based on 300 samples.

TABLE III

MATHEMATICAL PROGRAMMING EQUIVALENT FORMULATIONS

FOR ℓ_1, ℓ_2 and ℓ_∞ OPTIMIZATION

The original problem: minimize $H(\mathbf{f})$ over $\mathbf{x}$

The equivalent problem: minimize $V(\mathbf{x},\mathbf{y})$ over $\mathbf{x},\mathbf{y}$ subject to the constraints as defined below

$H(\mathbf{f})$	$V(\mathbf{x}, \mathbf{y})$	constraints (for $j = 1, 2, \ldots, m$)
$\|\mathbf{f}\|_1$	$\sum_{j=1}^{m} y_j$	$y_j \geq f_j,\ y_j \geq -f_j$
$\|\mathbf{f}\|_2$	$\mathbf{y}^T\mathbf{y}$	$y_j = f_j$
$\|\mathbf{f}\|_\infty$	y	$y \geq f_j,\ y \geq -f_j$
$H_1^+(\mathbf{f})$	$\sum_{j=1}^{m} y_j$	$y_j \geq f_j,\ y_j \geq 0$
$H_2^+(\mathbf{f})$	$\mathbf{y}^T\mathbf{y}$	$y_j \geq f_j,\ y_j \geq 0$
$H_\infty^+(\mathbf{f})$	y	$y \geq f_j, y \geq 0$
$H_\infty(\mathbf{f})$	y	$y \geq f_j$

Note: A generalized ℓ_p function $H_p(\mathbf{f})$ is defined through $H_p{}^+(\mathbf{f})$ and $H_p{}^-(\mathbf{f})$. $H_p{}^-$ is a continuously differentiable function for all $p < \infty$.

TABLE IV

SENSITIVITY EXPRESSIONS FOR SELECTED FREQUENCY RESPONSES

Response	Formula	Sensitivity Expression
input reflection coefficient ρ_{in}	$\frac{Y_S}{Y_S^*}[2\,G_S V_1 - 1]$	$2\left[V_1 \frac{\partial}{\partial\phi}\left(\frac{Y_S G_S}{Y_S^*}\right) + \frac{Y_S G_S}{Y_S^*}\frac{\partial V_1}{\partial\phi}\right]$
input return loss	$-20 \log_{10}\lvert\rho_{in}\rvert$	$-\frac{20}{\ln 10}\,\mathrm{Re}\left[\frac{1}{\rho_{in}}\frac{\partial\rho_{in}}{\partial\phi}\right]$
insertion loss	$-20 \log_{10}\lvert V_N Y_T\rvert$	$-\frac{20}{\ln 10}\,\mathrm{Re}\left[\frac{1}{V_N}\frac{\partial V_N}{\partial\phi} + \frac{1}{Y_T}\frac{\partial Y_T}{\partial\phi}\right]$
scattering matrix	$(\bar{z} - 1)(\bar{z} + 1)^{-1}$	$\frac{1}{2Z_0}(1 - S)\frac{\partial z}{\partial\phi}(1 - S)$

$G_S = \mathrm{Re}(Y_S)$ $Y_T = Y_S + Y_L$

Y_S^* is the complex conjugate of Y_S

$\bar{z} = z/Z_0$ where Z_0 is the normalizing impedance. S is the scattering matrix.

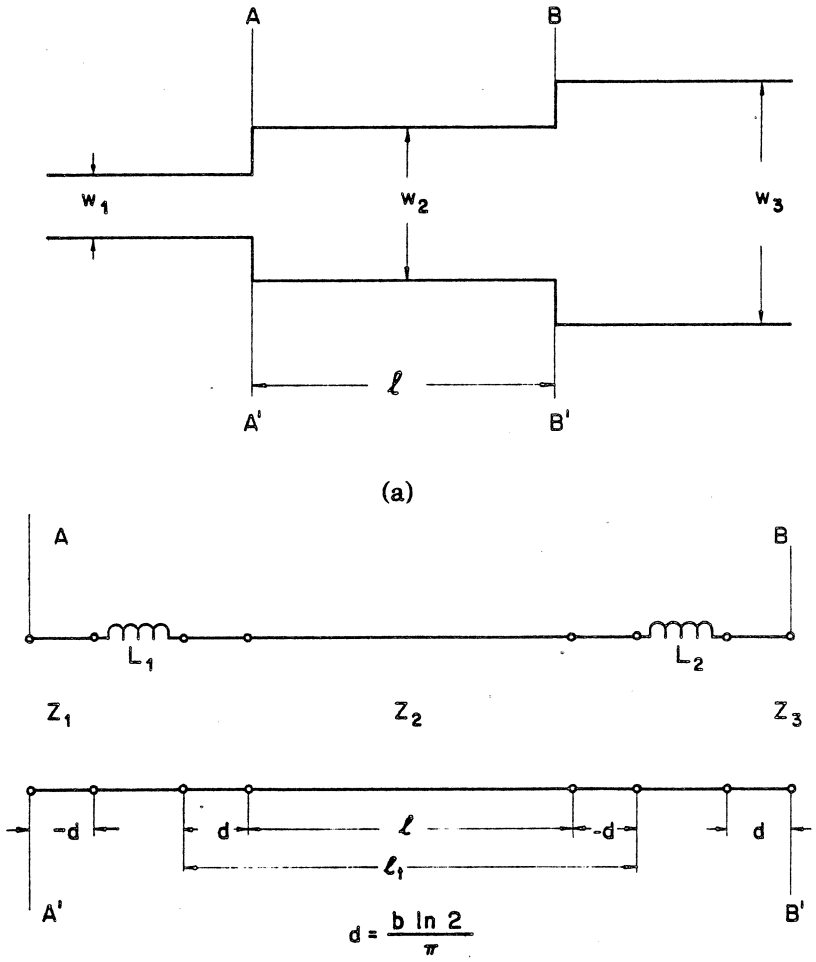


Fig. 1 A microwave stripline transformer showing (a) the physical structure and (b) the equivalent circuit model [30]. The physical parameters are

$$\Phi^M = [w_1 \;\; w_2 \;\; w_3 \;\; \ell \;\; \sqrt{\varepsilon_{r1}} \;\; \sqrt{\varepsilon_{r2}} \;\; \sqrt{\varepsilon_{r3}} \;\; b_1 \;\; b_2 \;\; b_3 \;\; t_{s1} \;\; t_{s2} \;\; t_{s3}]^T$$

where w is the strip width, ℓ the length of the middle section, ε_r the dielectric constant, b the substrate thickness and t_s the strip thickness. Φ^M is represented in the simulation model by Φ^L. The high-level parameters of the equivalent circuit are

$$\Phi^H = [D_1 \;\; D_2 \;\; D_3 \;\; L_1 \;\; L_2 \;\; \ell_t]^T$$

where D is the effective linewidth, L the junction parasitic inductance and ℓ_t the effective section length. Suitable empirical formulas that relate Φ^L to Φ^H can be found in [30].

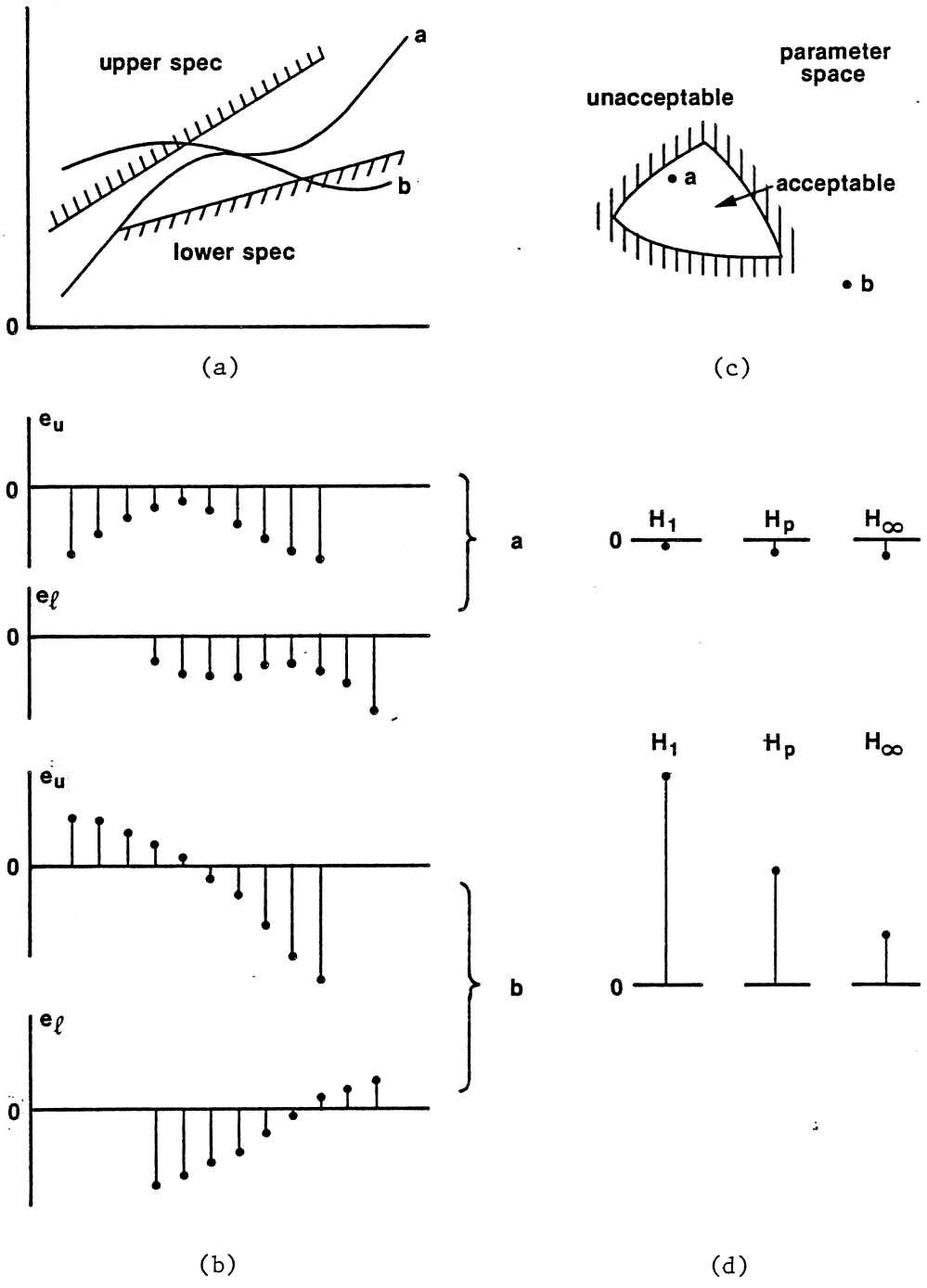


Fig. 2 Illustrations of (a) upper specifications, lower specifications and the responses of circuits a and b, (b) error functions corresponding to circuits a and b, (c) the acceptable region and (d) generalized ℓ_p objective functions defined in (13).

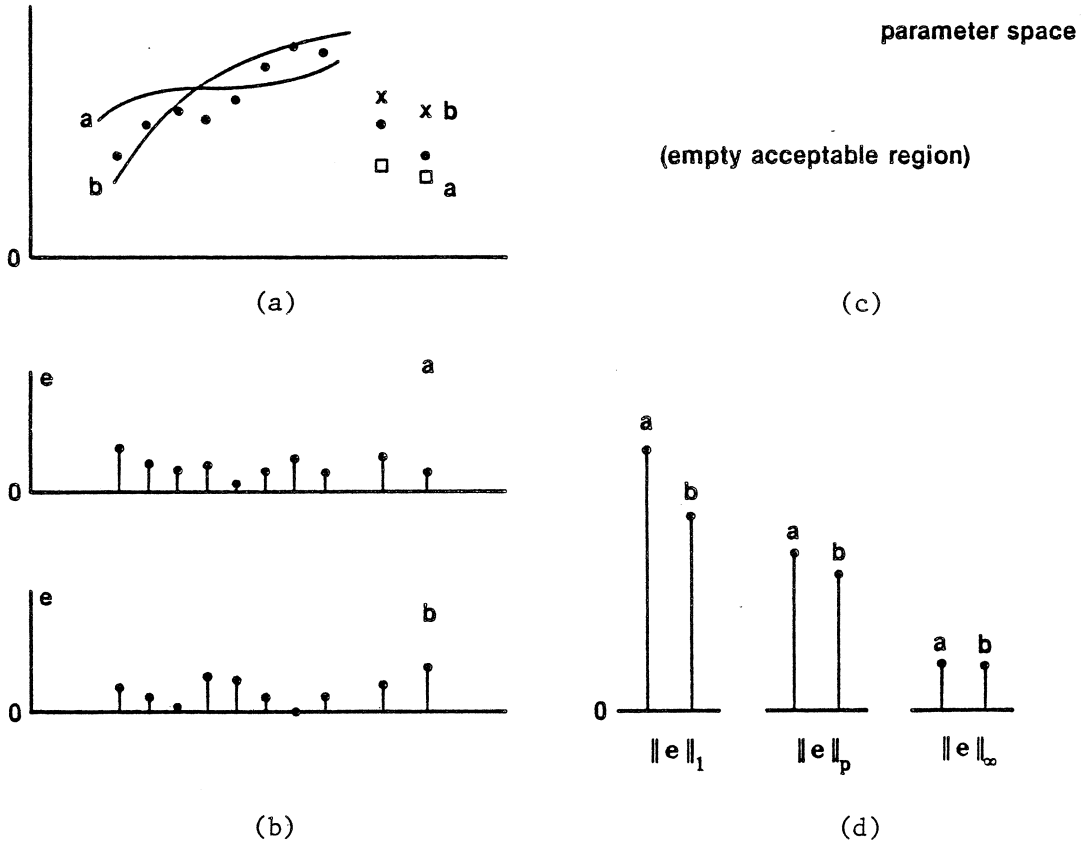


Fig. 3 Illustrations of (a) a discretized single specification and two discrete single specifications (e.g., expected parameter values to be matched), as well as the responses of circuits a and b, (b) error functions related to circuits a and b, (c) the (empty) acceptable region (i.e., a perfect match is not possible) and (d) the corresponding ℓ_p norms.

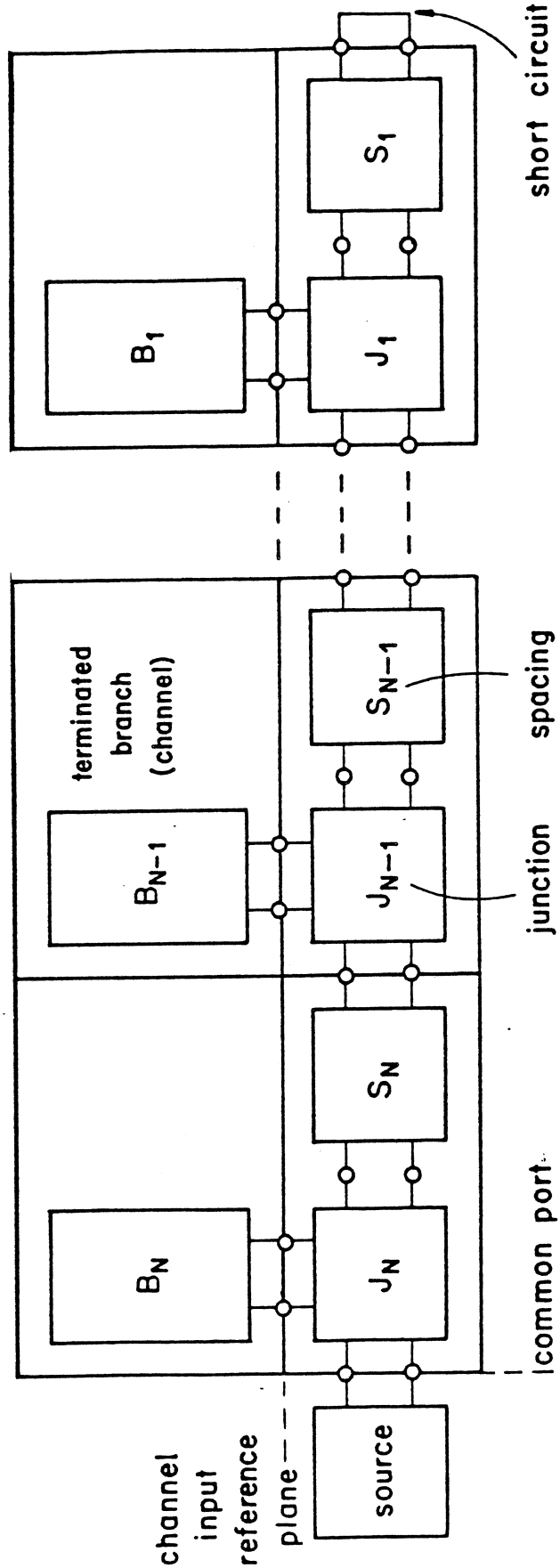


Fig. 4 A typical multiplexer structure. $J_1, J_2, \ldots, J_N$ are junction models, $B_1, B_2, \ldots, B_N$ are terminated channels and $S_1, S_2, \ldots, S_N$ represent spacings along the manifold.

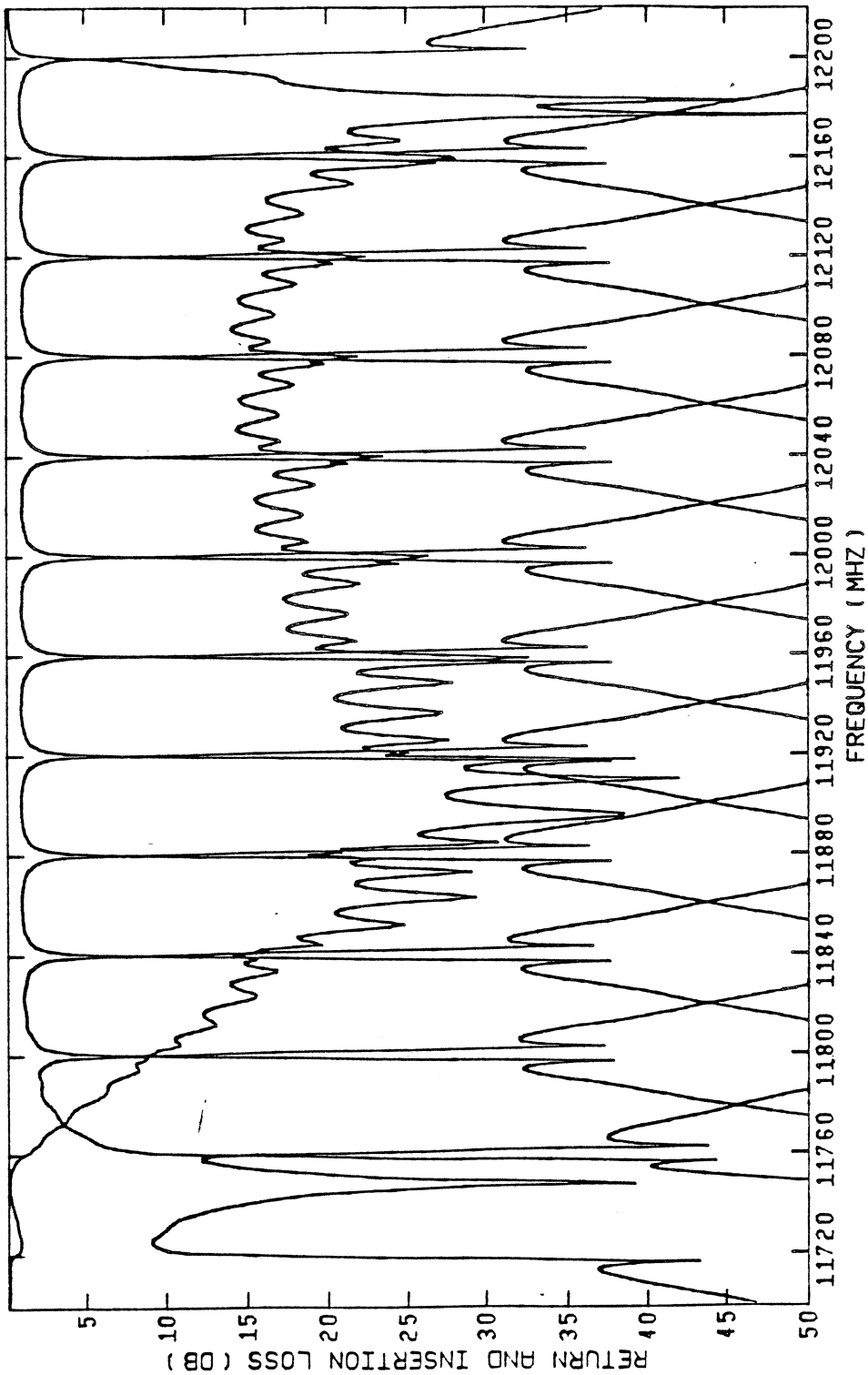


Fig. 5 Responses of the 12 channel multiplexer at the start of the optimization process.

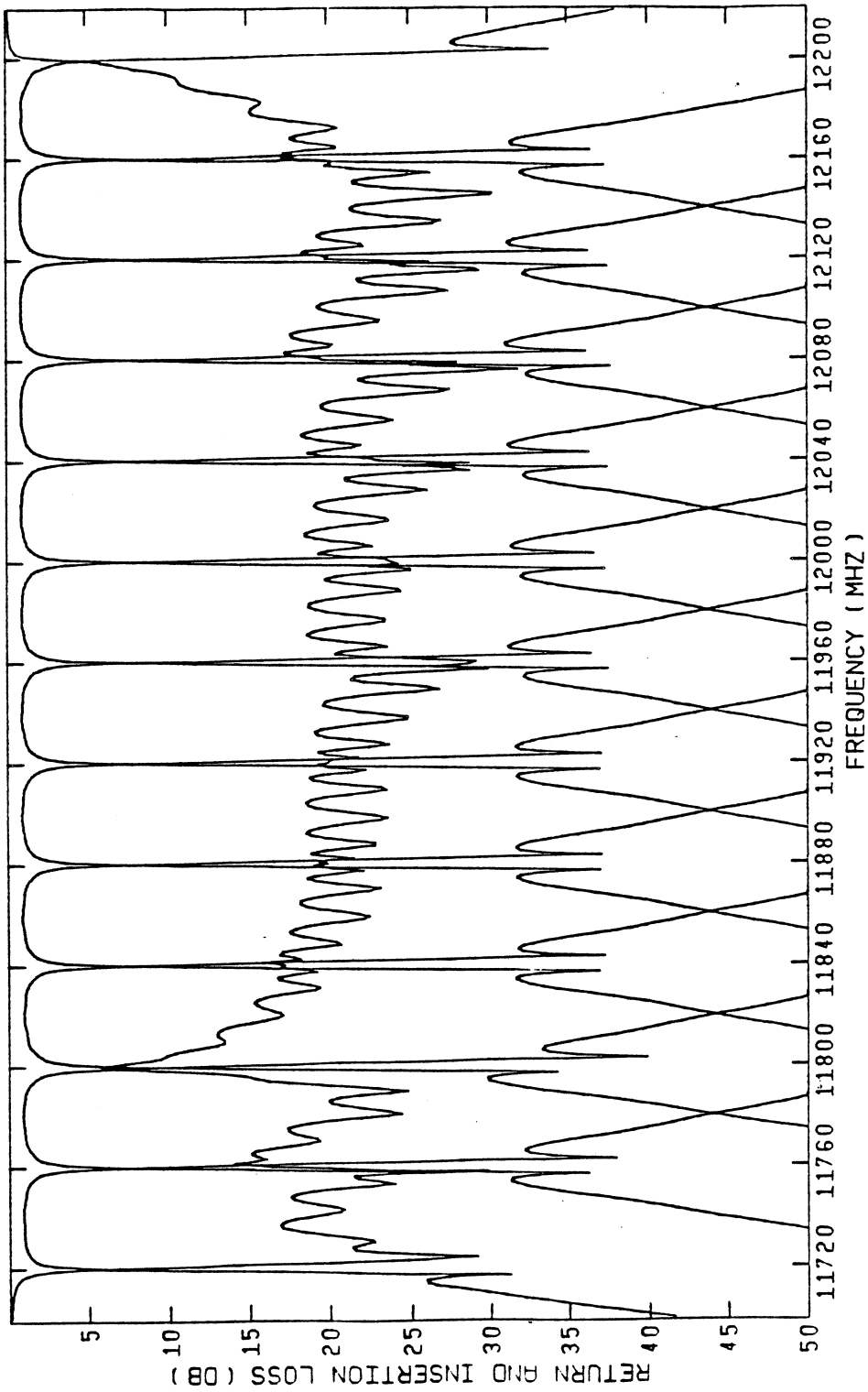


Fig. 6 Responses of the 12 channel multiplexer with optimized spacings using ℓ_1 optimization.

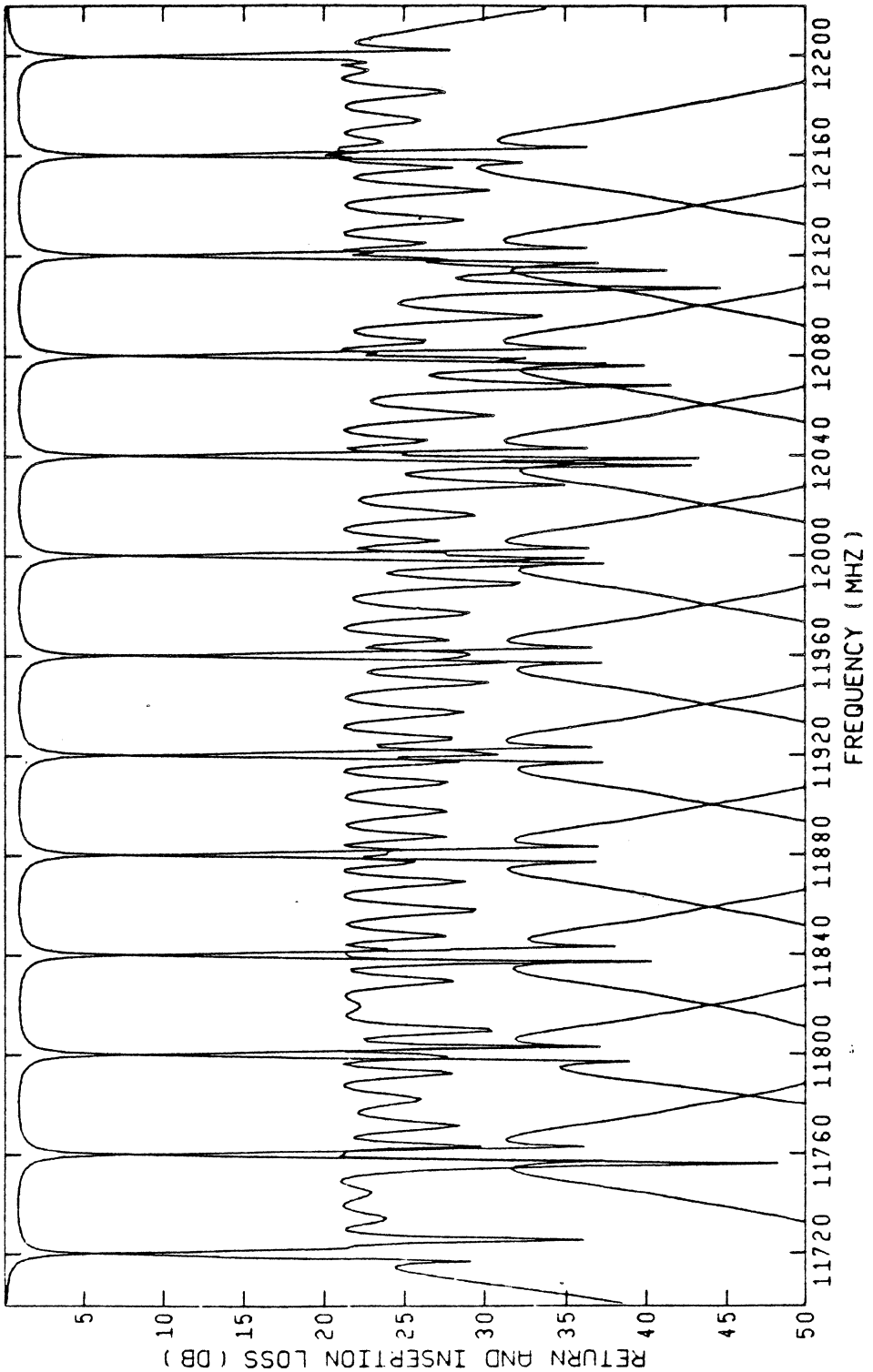


Fig. 7 Responses of the 12 channel multiplexer with optimized spacings, couplings and resonances using minimax optimization.

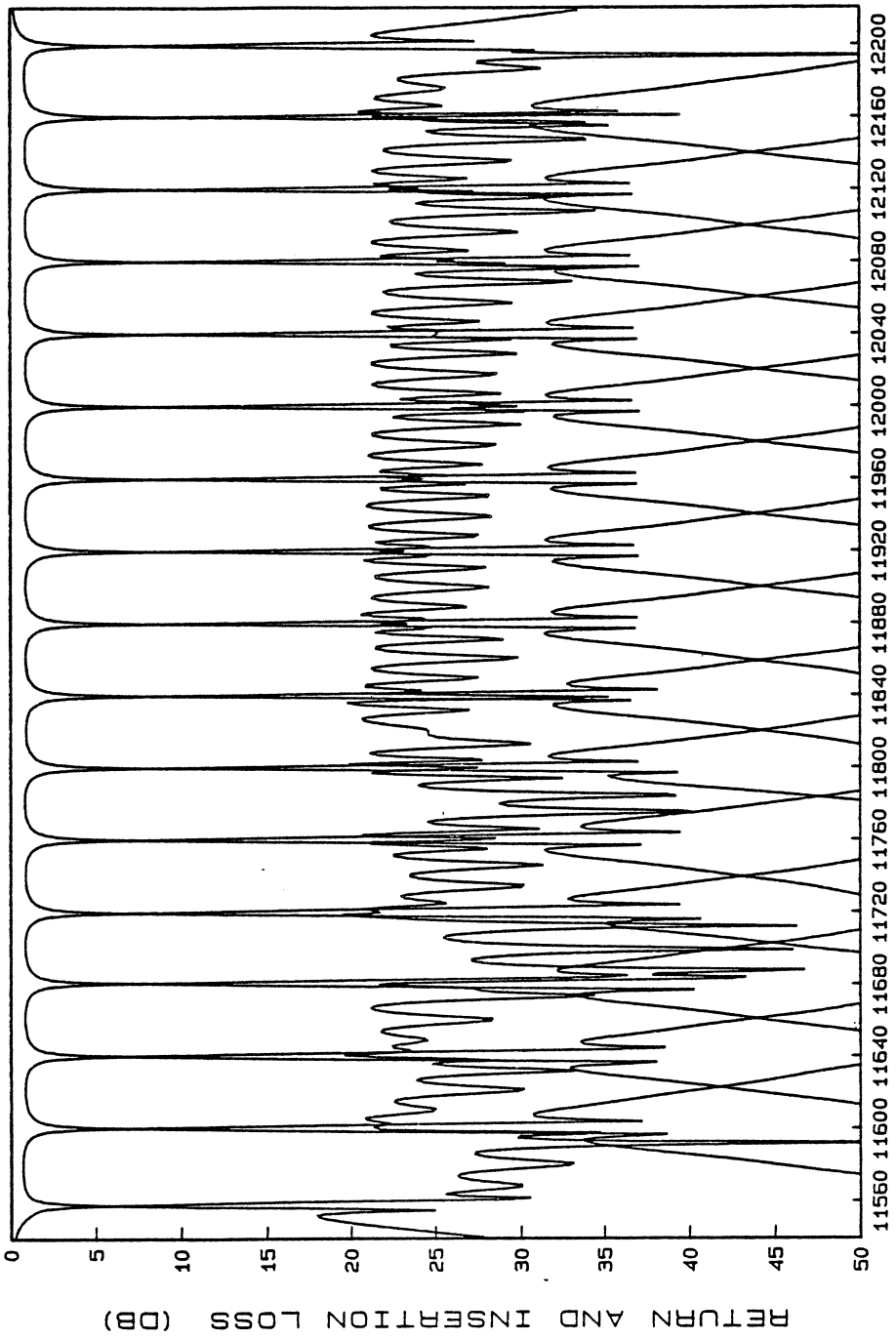


Fig. 8 Optimized responses of the 16 channel multiplexer.

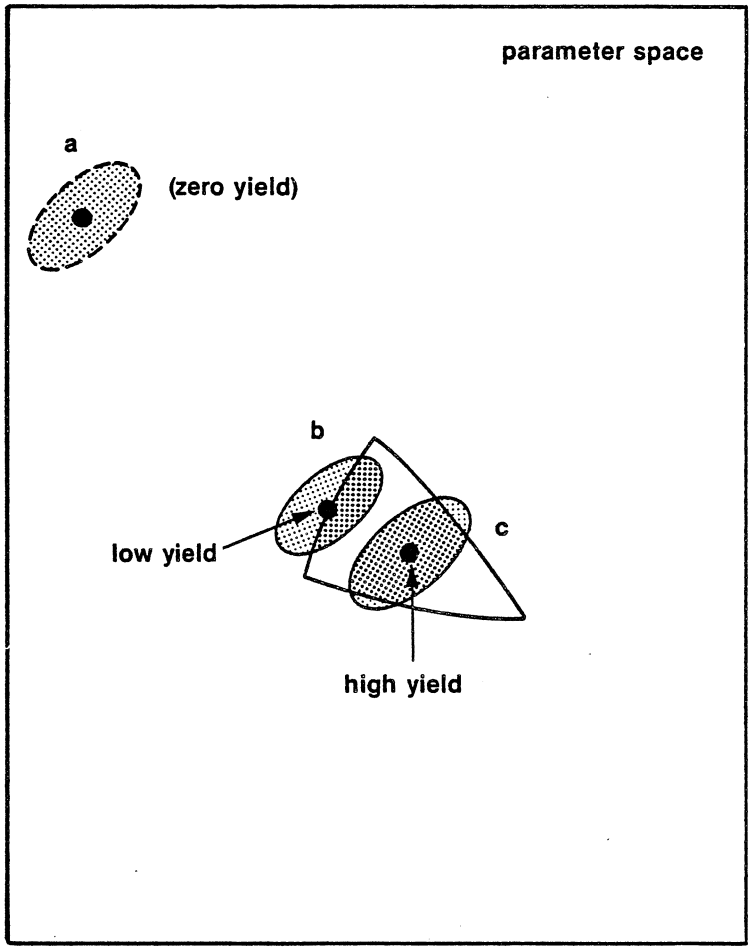


Fig. 9 Three nominal points and the related yield.

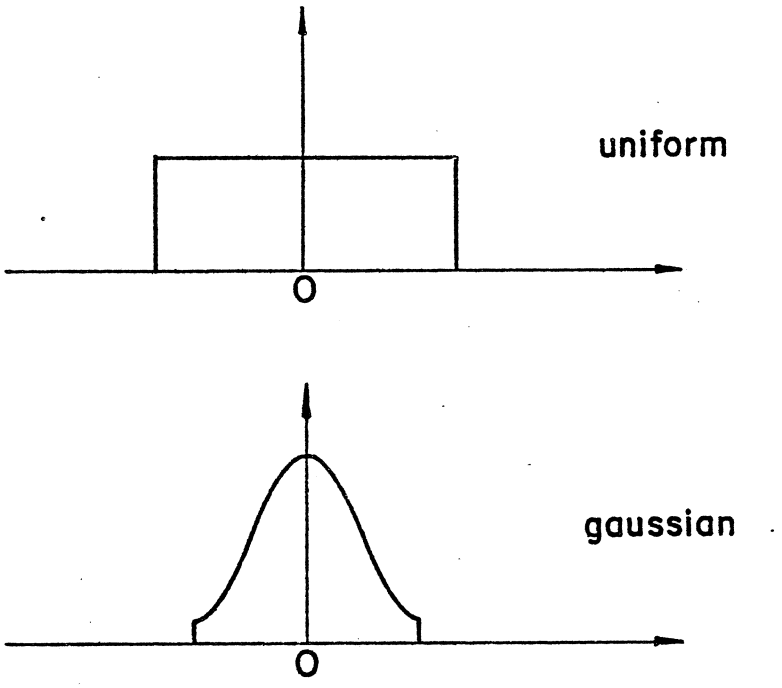
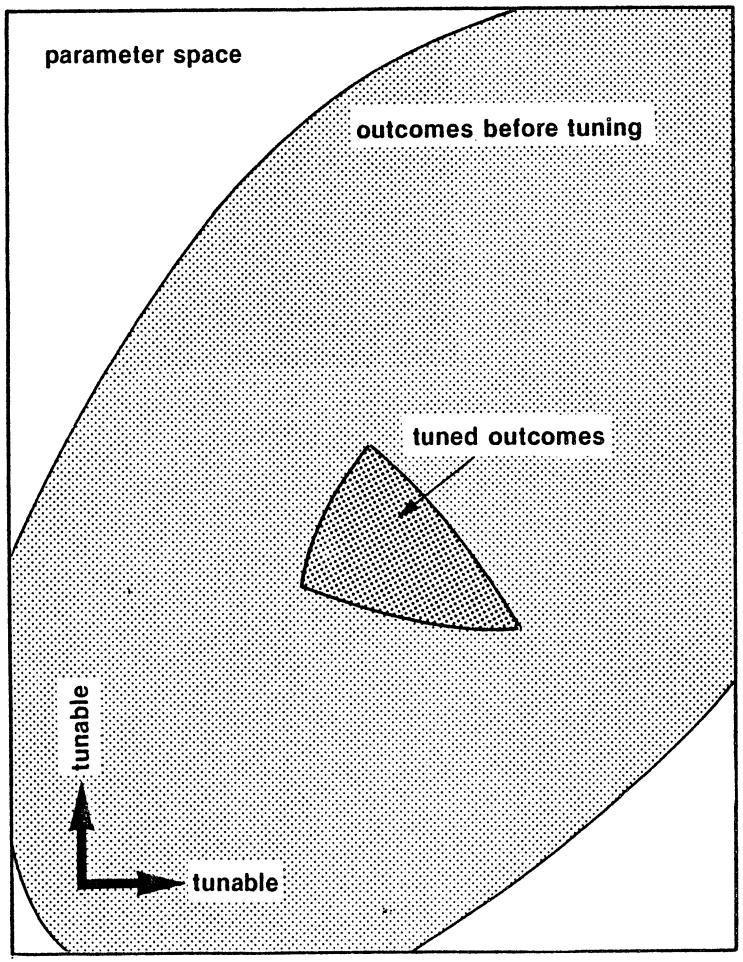
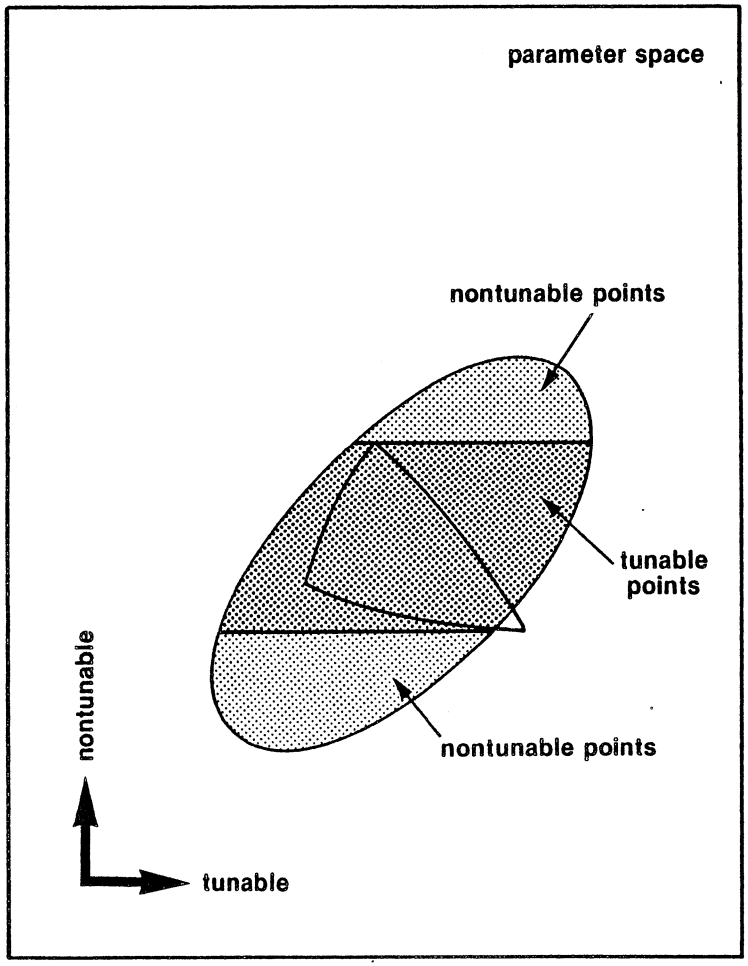


Fig. 10 Typical tolerance distributions: uniform and Gaussian (normal).



(a)



(b)

Fig. 11 Illustrations of tuning: (a) both parameters are tunable for a case in which the probability that an untuned design meets the specifications is very low and (b) only one parameter is tunable.

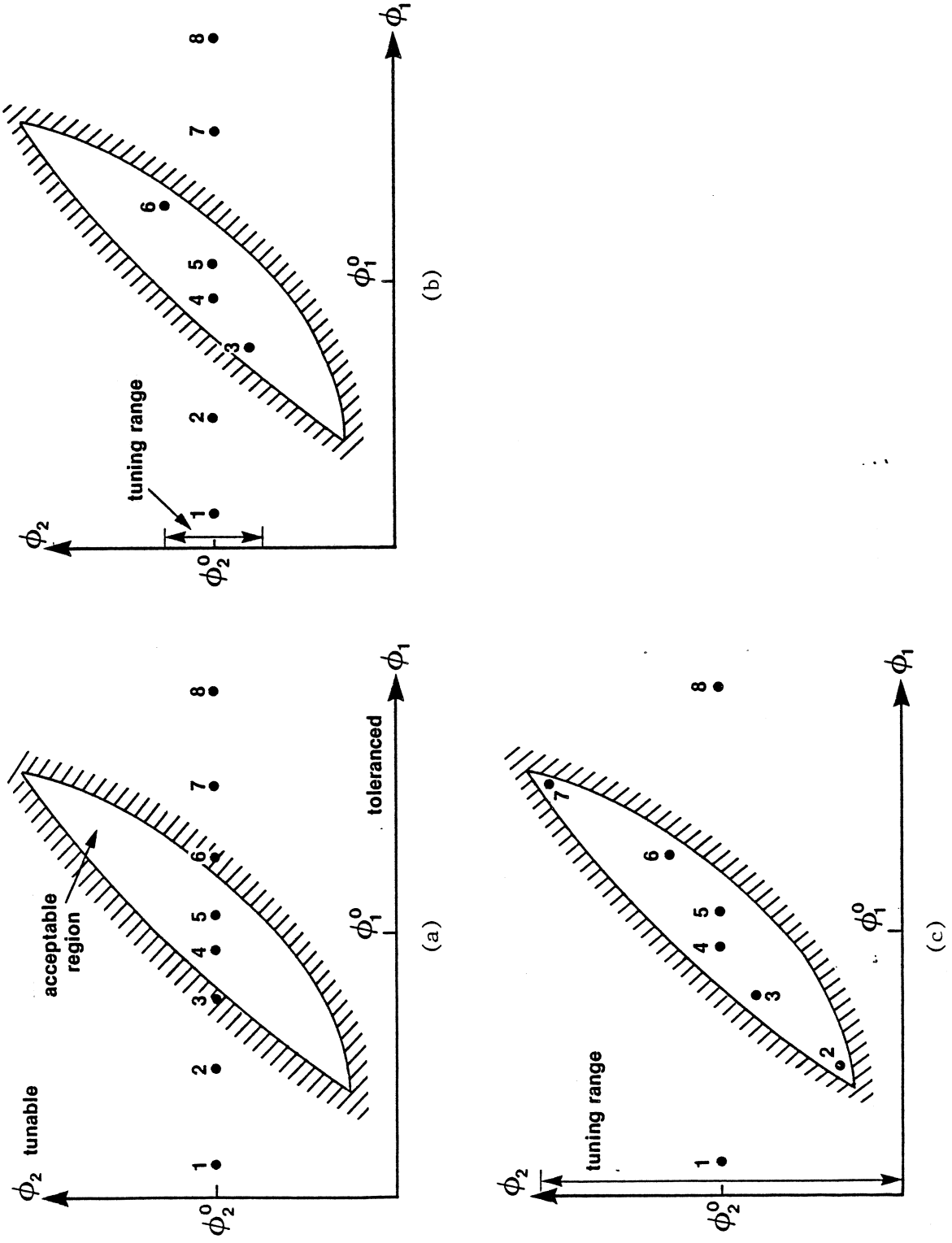


Fig. 12 An illustration of multi-circuit design considering eight circuit outcomes. Φ_1 is toleranced and Φ_2 is tunable.

(a) Without tuning the yield is 2/8 (25%).
(b) Tuning on Φ_2 is restricted to a small range. The improved yield is 4/8 (50%).
(c) A 75% yield is achieved by allowing a large tuning range.

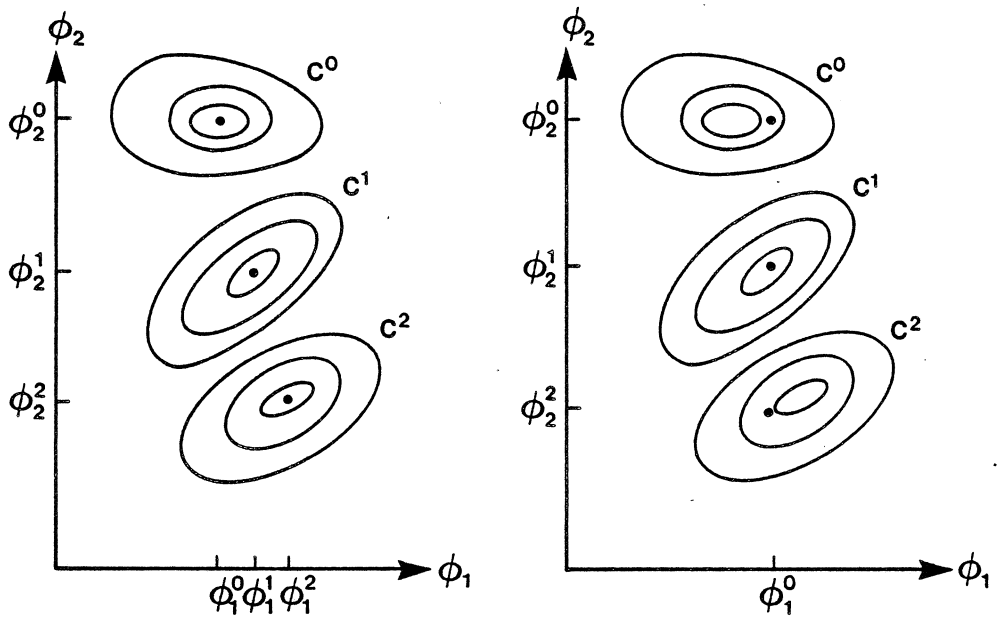


Fig. 13 An illustration of multi-circuit modeling. Three circuits are created by making two physical adjustments. Assume that we know that ϕ_1 should not be affected by the physical adjustments. C^0, C^1 and C^2 are contours of the error functions corresponding to the three circuits.
(a) By treating the three circuits separately, we obtain Φ^0, Φ^1 and Φ^2. ${\phi_1}^0$, ${\phi_1}^1$ and ${\phi_1}^2$ turn out to have different values (which is inconsistent with our knowledge) because of uncertainties.
(b) Consistent results can be obtained by defining ϕ_1 as a common variable and processing three circuits simultaneously.

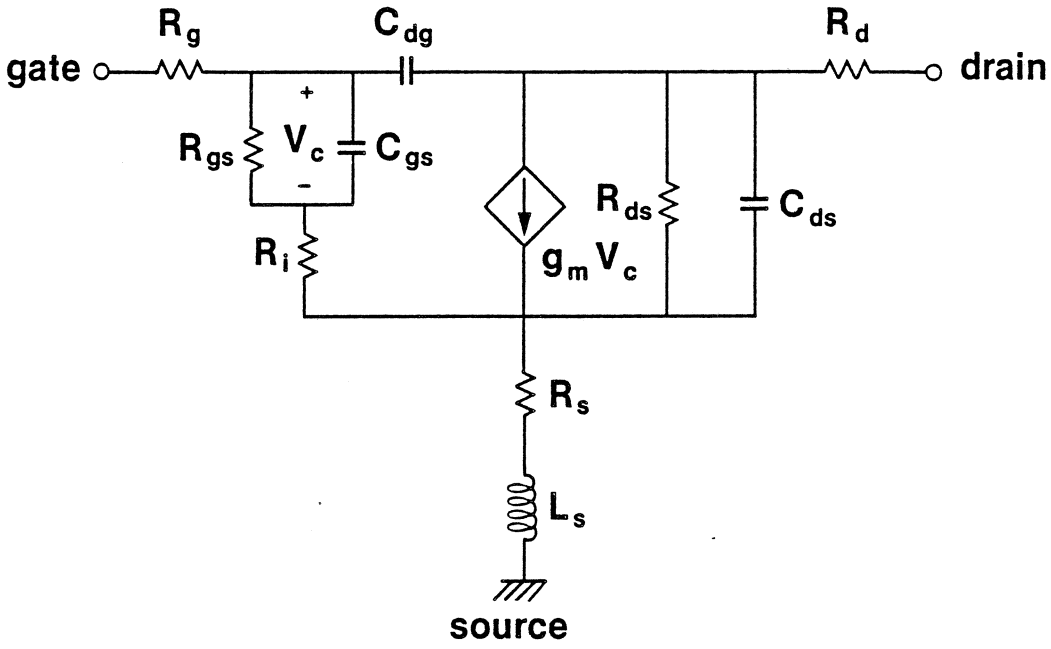
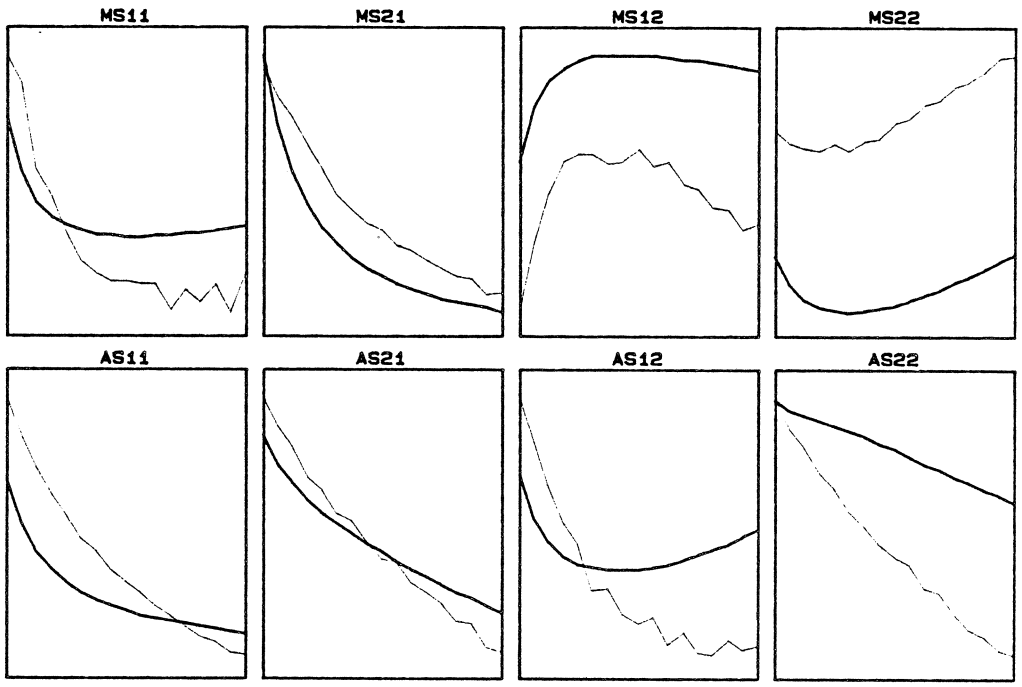
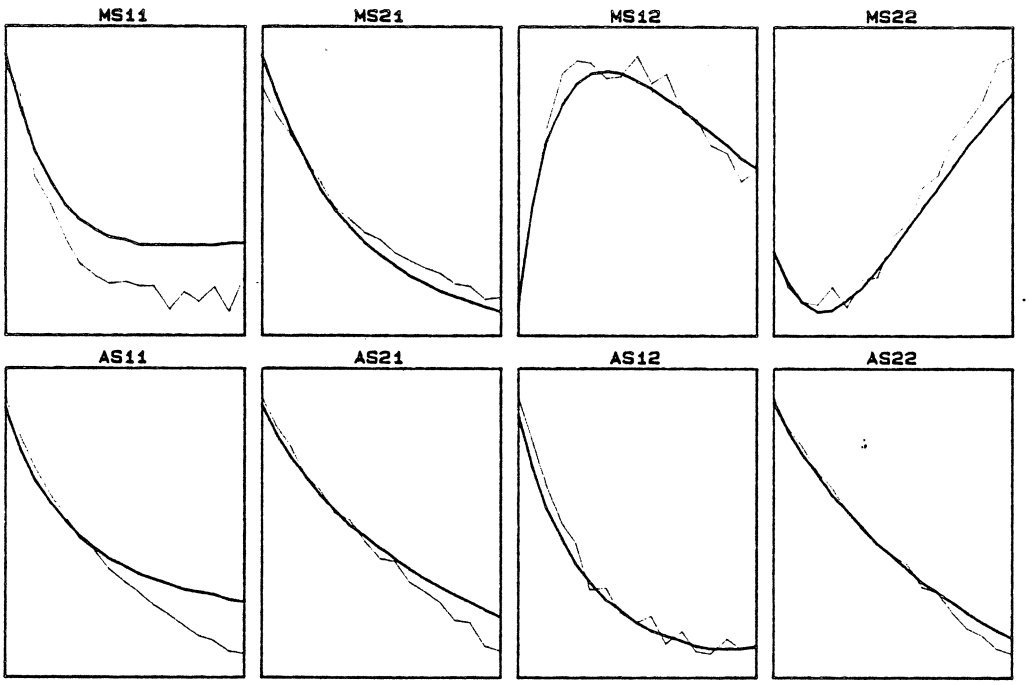


Fig. 14 The small signal equivalent circuit model for a FET device [21].



(a)



(b)

Fig. 15 The scattering parameter match [21] between the model and the measurements at (a) the starting point and (b) the solution. The biasing conditions are V_{ds} = 4V, V_{gs} = 0V and I_{ds} = 177mA.

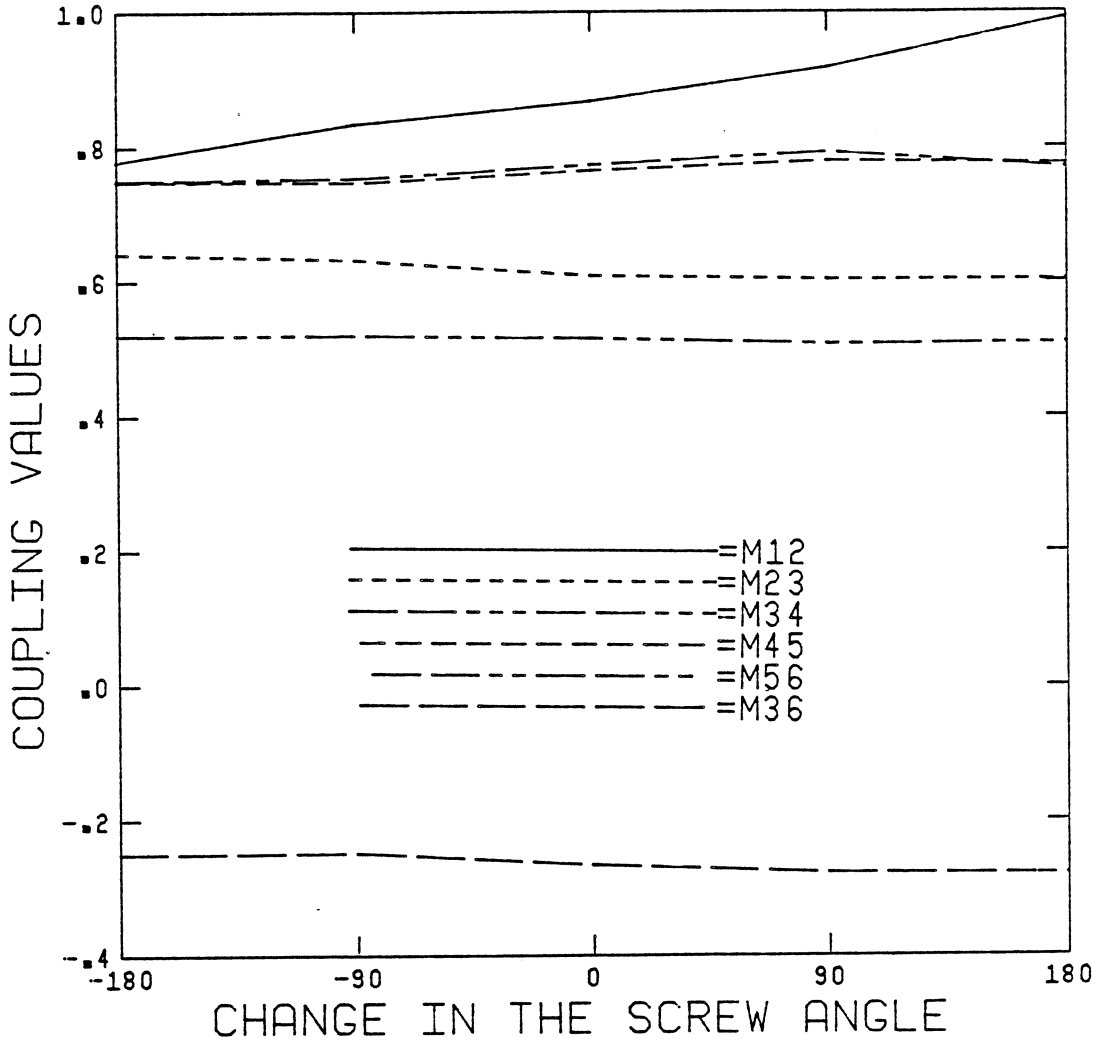


Fig. 16 The identified coupling values versus the relative position of the screw in degrees for the 6th order filter example. The screw was assumed to control M_{12}.

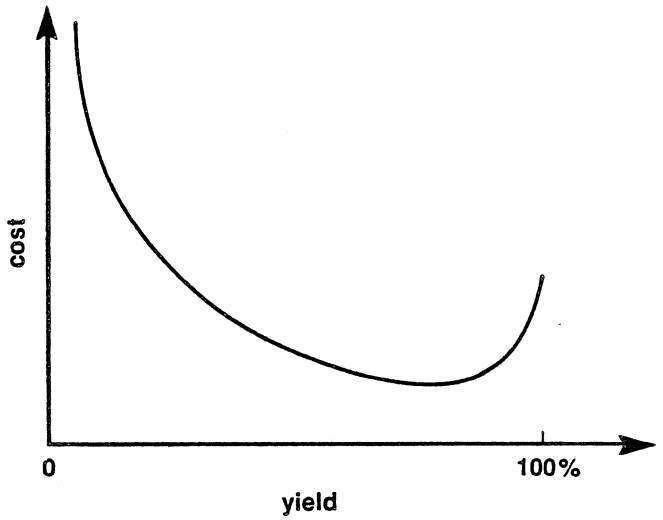


Fig. 17 A typical cost-versus-yield curve [111].

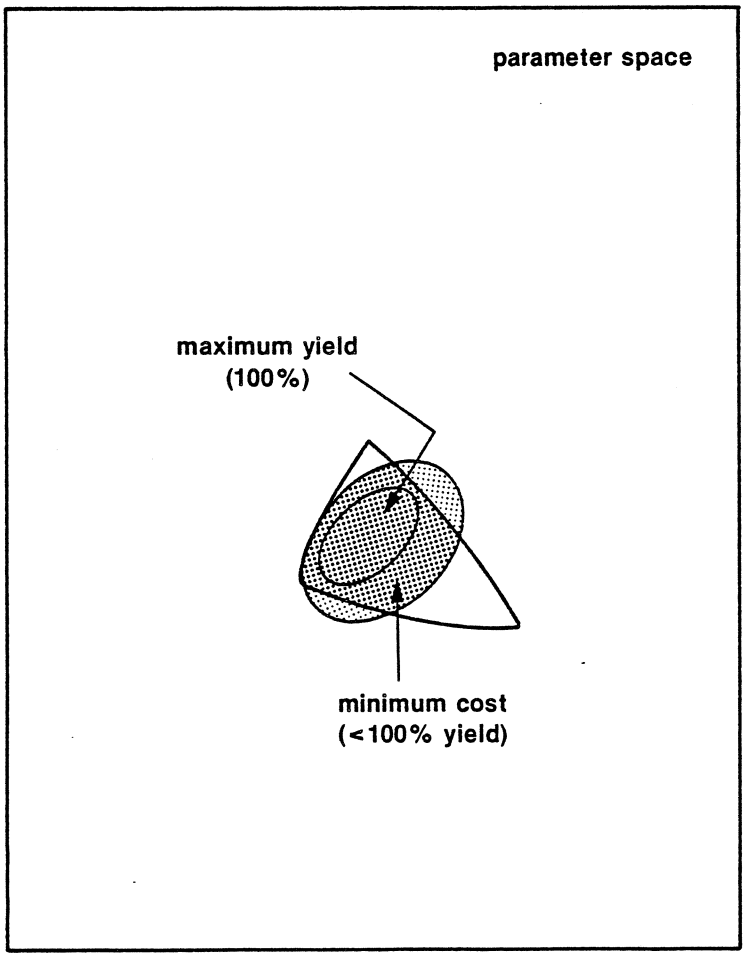


Fig. 18 A maximum yield design and a minimum cost design.

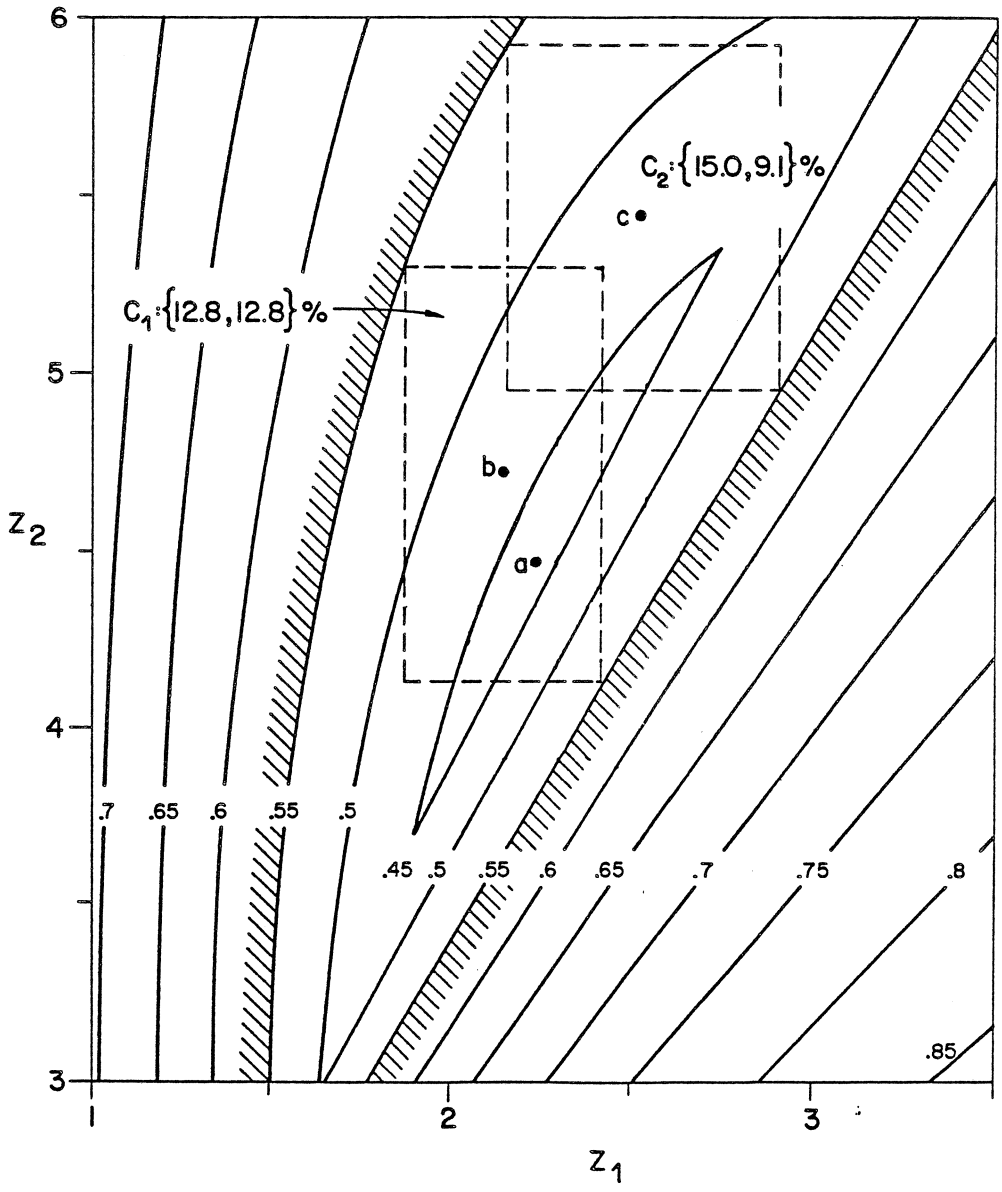


Fig. 19 Contours of max $|\rho_j|$ with respect to Z_1 and Z_2 for the two-section transformer indicating the minimax nominal solution a, the centered design with relative tolerances b and the centered design with absolute tolerances c. The values in brackets are the optimized tolerances (as percentages of the nominal values). The specification is $|\rho| \leq 0.55$.

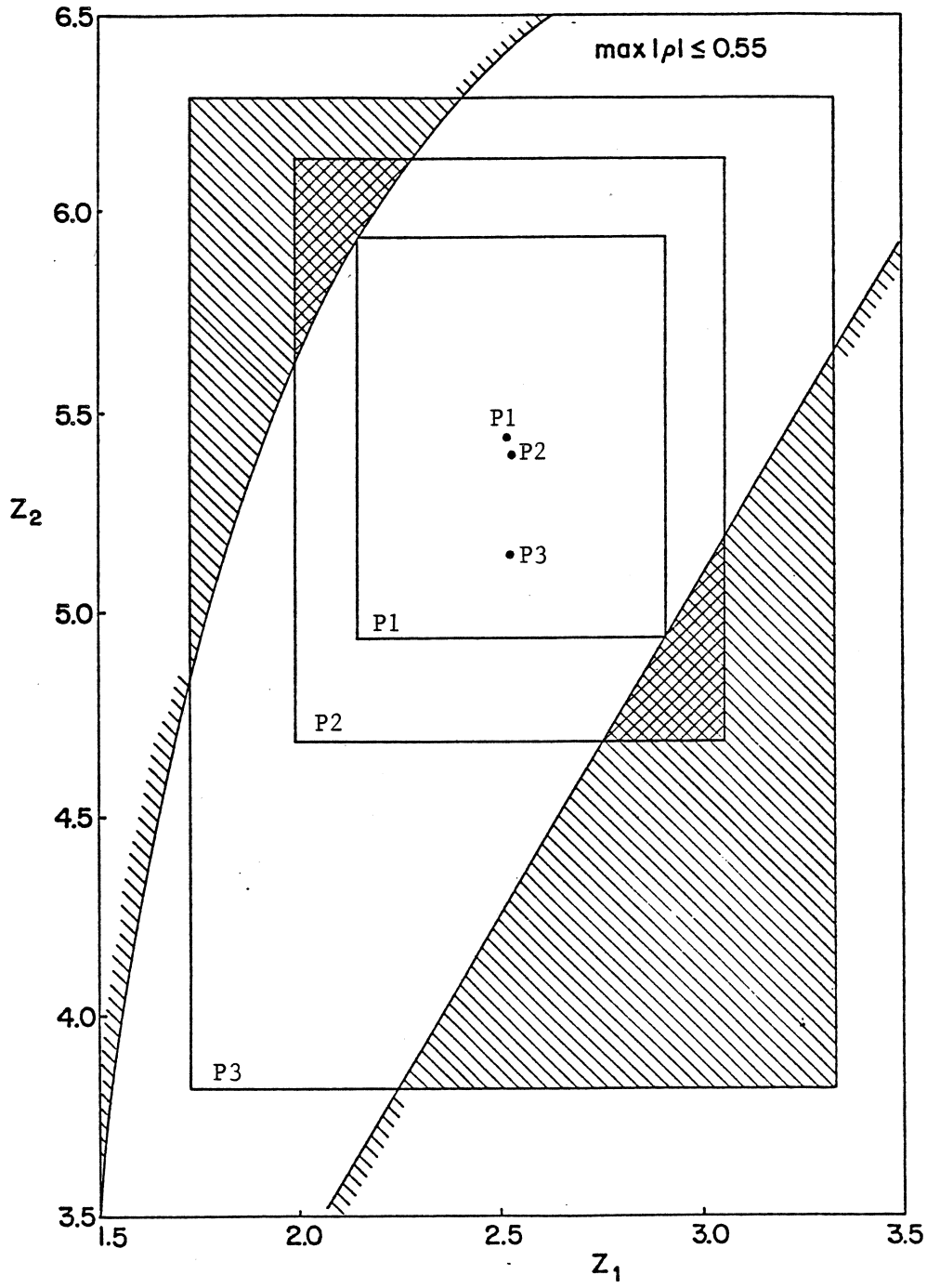


Fig. 20 The optimized tolerance regions and nominal values for the worst case design P1, 90% yield design P2 and minimum cost design P3 of the two-section transformer.

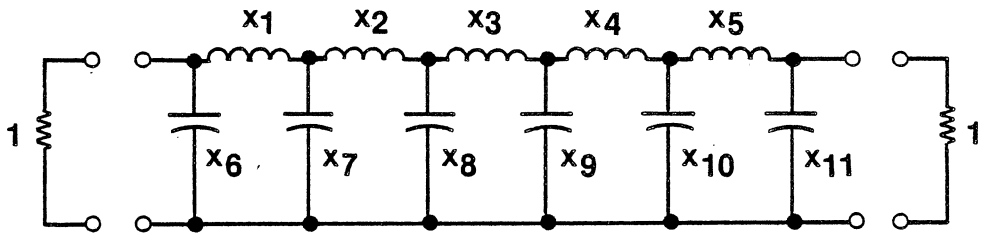


Fig. 21 The Chebyshev lowpass filter [111].