OPTIMIZATION
OF MICROWAVE CIRCUITS:
TECHNIQUES FOR THE 1990's

OSA-87-MC-10-R, September 1987

OPTIMIZATION OF MICROWAVE CIRCUITS:

TECHNIQUES FOR THE 1990's

This is a collection of three papers to be published in the IEEE Transactions on Microwave Theory and Techniques. The first paper [1], by Bandler and Chen, is a state of the art review. The second paper [2], by Bandler and Zhang, addresses the optimization of large microwave circuits using novel decomposition techniques. The third paper [3], by Bandler, Chen, Daijavad and Madsen, concerns optimization algorithms not requiring user-supplied derivatives.

References [1] and [3] will appear in the Special Issue on Computer-Aided Design in February 1988. Reference [2] is to appear in the 1987 Symposium Issue in December 1987.

The set of three papers describes theoretical and practical advances that will be essential for microwave integrated circuit design for the 1990's. Optimization Systems Associates Inc. is currently developing software systems incorporating these approaches.

[1] J.W. Bandler and S.H. Chen, "Circuit optimization: the state of the art," IEEE Trans. Microwave Theory Tech., vol. MTT-36, 1988.

[2] J.W. Bandler and Q.J. Zhang, "An automatic decomposition approach to optimization of large microwave systems," IEEE Trans. Microwave Theory Tech., vol. MTT-35, 1987.

[3] J.W. Bandler, S.H. Chen, S. Daijavad and K. Madsen, "Efficient optimization with integrated gradient approximations," IEEE Trans. Microwave Theory Tech., vol. MTT-36, 1988.

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. 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. Illustrative examples as well as a comprehensive bibliography are provided.

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

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 [102] 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 [73,74,108] demonstrated optimal design of linear arrays and filters using the penalty function approach. Two papers in 1969 by Director and Rohrer [48,49] 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 [4,5] 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 [52] and Karafin [68] 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 [37], Charalambous [39], Bandler and Rizk [26], Hachtel and Sangiovanni-Vincentelli [63], and Brayton et al. [32] 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, as well as recent gradient-based optimization methods

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 [83]), the more esoteric generalized ℓ_p objective (Charalambous [41]) or the minimax objective (Madsen et al. [80]). 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 [94] and Laker et al. [71]. 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 [6], Tromp [107]). 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. [23,24]), 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 [1,26,30,47,97,98,100,101] 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. [12]).

Our presentation is outlined as follows.

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.

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. [24], Polak [89]), simplicial (Director and Hachtel [47]) and multi-dimensional (Bandler and Abdel-Malek [7]) approximations of the acceptable region, the gravity method (Soin and Spence [98]) and the parametric sampling method (Singhal and Pinel [97]). 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 gradient-based ℓ_p optimization methods is reviewed in Section VII. Emphasis is given to the trust region Gauss-Newton and the quasi-Newton algorithms (Madsen [78], Moré [82], Dennis and Moré [46]).

The subject of gradient calculation and approximation is briefly discussed in Section VIII.

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 $\boldsymbol{\phi}^M$.

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 $\mathbf{F}^M(\boldsymbol{\phi}^M)$.

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 [106,107] has considered an arbitrary number of levels (also see Bandler et al. [19]). Here, for simplicity, we consider a hierarchy of models consisting of four typical levels as

$$\begin{aligned} \mathbf{F}^{H} &= \mathbf{F}^{H}(\mathbf{F}^{L}), \\ \mathbf{F}^{L} &= \mathbf{F}^{L}(\boldsymbol{\phi}^{H}), \\ \boldsymbol{\phi}^{H} &= \boldsymbol{\phi}^{H}(\boldsymbol{\phi}^{L}). \end{aligned} \qquad (1)$$

$\boldsymbol{\phi}^L$ 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., $\boldsymbol{\phi}^M$. $\boldsymbol{\phi}^H$ 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 $\boldsymbol{\phi}^L$ and $\boldsymbol{\phi}^H$ 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 $\mathbf{F}^L$, 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. [25]. 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 [5], and Bandler and Rizk [26], 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,

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

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

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

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

$$e_{uj}(\boldsymbol{\phi}) = w_{uj}(F_j(\boldsymbol{\phi}) - S_{uj}) \;, \qquad j \in J_u \,,$$
$$e_{\ell j}(\boldsymbol{\phi}) = w_{\ell j}(F_j(\boldsymbol{\phi}) - S_{\ell j}) \;, \qquad j \in J_\ell \,. \tag{4}$$

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

$$J_u = \{j_1, j_2, ..., j_k\} \;,$$
$$J_\ell = \{j_{k+1}, j_{k+2}, ..., j_m\} \,. \tag{5}$$

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

$$e_i = e_{uj}(\boldsymbol{\phi}) \;, \qquad j = j_i \,, \quad i = 1, 2, ..., k \,,$$
$$e_i = -e_{\ell j}(\boldsymbol{\phi}), \qquad j = j_i \,, \quad i = k+1, k+2, ..., m \,. \tag{6}$$

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 $\boldsymbol{\phi}$ which may have been normalized or transformed, it may include some statistical variables of interest, several parameters in $\boldsymbol{\phi}$ may be tied to one variable in $\mathbf{x}$, and so on.

Typically, the objective function U(**x**) is closely related to an ℓ_p norm or a generalized ℓ_p function of **e**($\boldsymbol{\phi}$). 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 [103]) of **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 [83]), 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 earlier versions of the commercial CAD packages TOUCHSTONE [104] and SUPER-COMPACT [99] have 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 = ∞ we have the minimax norm

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

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 [3,13,14,21,40,42,65,67,80,85].

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

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

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 [22,29,66,86] and, more recently, to fault location [8,9,27] and robust device modelling [12].

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

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

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

Bandler and Charalambous [10,41] have proposed the use of a generalized ℓ_p function defined by

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

where

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

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.

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

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

The current Version 1.5 of TOUCHSTONE [105] offers the generalized ℓ_p optimization techniques, including minimax.

The Acceptable Region

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

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

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

III. NOMINAL CIRCUIT OPTIMIZATION

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 $\boldsymbol{\phi}^0$, which satisfies the specifications Furthermore, if we consider the functional relationship of $\boldsymbol{\phi}^H = \boldsymbol{\phi}^H(\boldsymbol{\phi}^L)$ 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 $\boldsymbol{\phi}^{H,0}$ is obtained through the use of an equivalent circuit and/or a transfer function. A low-level model $\boldsymbol{\phi}^{L,0}$ is then calculated from $\boldsymbol{\phi}^{H,0}$, 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 $\boldsymbol{\phi}^{L,0}$ by its physical counterpart $\boldsymbol{\phi}^{M,0}$.

With the tool of mathematical optimization, the nominal point $\boldsymbol{\phi}^0$ (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(e). The vector $\mathbf{x}$ contains all the elements of or a subset of the elements of $\boldsymbol{\phi}^0$. 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. The minimax optimization of manifold multiplexers as described by Bandler et al. [18,22,28] provides an excellent illustration of large-scale nominal design of microwave circuits.

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

$$\mathbf{F}^{M} = \mathbf{F}^{M,0}(\boldsymbol{\phi}^{M}) + \Delta\mathbf{F}^{M} ,$$
$$\boldsymbol{\phi}^{M} = \boldsymbol{\phi}^{M,0} + \Delta\boldsymbol{\phi}^{M} , \qquad (17)$$

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

$$\mathbf{F}^{H} = \mathbf{F}^{H,0}(\mathbf{F}^{L}) + \Delta\mathbf{F}^{H} ,$$
$$\mathbf{F}^{L} = \mathbf{F}^{L,0}(\boldsymbol{\phi}^{H}) + \Delta\mathbf{F}^{L} , \qquad (18)$$
$$\boldsymbol{\phi}^{H} = \boldsymbol{\phi}^{H,0}(\boldsymbol{\phi}^{L}) + \Delta\boldsymbol{\phi}^{H} ,$$
$$\boldsymbol{\phi}^{L} = \boldsymbol{\phi}^{L,0} + \Delta\boldsymbol{\phi}^{L} .$$

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 [25] 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 $\Delta\boldsymbol{\phi}^{M}$ and are represented in the model by $\Delta\boldsymbol{\phi}^{L}$. We may also use $\Delta\boldsymbol{\phi}^{H}$ 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 $\Delta\mathbf{F}^{H}$ ($\mathbf{F}^{H}$ being the actual reflection coefficient; see [25] 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 $\boldsymbol{\phi}^H$), the uncertainty associated with r may appear in $\Delta\boldsymbol{\phi}^H$ or in $\Delta\mathbf{F}^L$.

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 $\Delta\boldsymbol{\phi}^L$ and model uncertainties $\Delta\boldsymbol{\phi}^H$. First of all, we represent a realistic situation by multiple circuits as

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

where $\boldsymbol{\phi}^0$, $\boldsymbol{\phi}^k$ and $\mathbf{s}^k$ 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

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

where $H(\mathbf{e}) \le 0$, defined in (13), indicates satisfaction of the specifications by $\boldsymbol{\phi}$. An estimate of the yield is given by the percentage of acceptable samples out of the total, as

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

The merit of a design can then be judged more realistically according to the yield it promises, as illustrated in Fig. 4. Now we shall have a closer look at the definition of multiple circuits.

In the Monte Carlo method the deviates $\mathbf{s}^k$ are constructed by generating random numbers using a physical process or arithmetical algorithms. Typically, we assume a statistical distribution for $\Delta\boldsymbol{\phi}^L$, denoted by $D^L(\boldsymbol{\varepsilon}^L)$ where $\boldsymbol{\varepsilon}^L$ 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. 5.

At the low level, consider

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

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

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

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. Incidentally, the nominal optimization problem, i.e., the traditional design problem, is sometimes referred to as the zero tolerance problem (ZTP).

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).

Tuning some components of $\boldsymbol{\phi}^M$ 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 $\boldsymbol{\tau}^k$ for each circuit, as

$$\boldsymbol{\phi}^k = \boldsymbol{\phi}^0 + \mathbf{s}^k + \boldsymbol{\tau}^k, \qquad k = 1, 2, \ldots, K. \tag{25}$$

We have to determine, through optimization, the value of $\boldsymbol{\tau}^k$ such that the specifications will be satisfied at $\boldsymbol{\phi}^k$ which may otherwise be unacceptable, as depicted in Figs. 6 and 7. 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 $\boldsymbol{\tau}^k$ 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 [20].

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. [12] can be used to overcome these difficulties. Multiple circuits are created by making deliberate adjustments on the physical parameters $\boldsymbol{\phi}^{M}$. 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 $\boldsymbol{\phi}$ to change by an unknown amount. For this approach to be successful, each physical adjustment should produce changes in only a few parameters in $\boldsymbol{\phi}$.

Although we do not know the changes in $\boldsymbol{\phi}$ 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 ([12], Section III.A) in which $\boldsymbol{\phi}$ can not be uniquely identified due to inaccessible nodes. The problem was effectively addressed using the multi-circuit approach.

To formulate this mathematically, let

$$\boldsymbol{\phi}^k = \begin{bmatrix} \boldsymbol{\phi}_c^k \\ \boldsymbol{\phi}_a^k \end{bmatrix}, \tag{26}$$

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

$$\mathbf{x} = [\,(\boldsymbol{\phi}^0)^T \;\; (\boldsymbol{\phi}_a^1)^T \ldots (\boldsymbol{\phi}_a^K)^T\,]^T \;, \tag{27}$$

and state the optimization problem as to

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

where

$$\mathbf{f} = [\mathbf{e}^T(\boldsymbol{\phi}^0) \;\; \mathbf{e}^T(\boldsymbol{\phi}^1) \;\ldots\; \mathbf{e}^T(\boldsymbol{\phi}^K)]^T \,. \tag{29}$$

Although any ℓ_p norm may be used, the unique property of ℓ_1 discussed in detail by Bandler et al. [12] can be exploited to great advantage. The concept of common and independent variables is depicted in Fig. 8.

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

$$\mathbf{x} = [\,(\boldsymbol{\phi}^0)^T \;\; (\boldsymbol{\phi}^1)^T \ldots (\boldsymbol{\phi}^K)^T\,]^T \;, \tag{30}$$

and change the objective function to an ℓ_p norm of

$$\mathbf{f} = [\mathbf{e}^T(\boldsymbol{\phi}^0) \ldots \mathbf{e}^T(\boldsymbol{\phi}^K) \quad \alpha_1(\boldsymbol{\phi}^1 - \boldsymbol{\phi}^0)^T \ldots \alpha_K(\boldsymbol{\phi}^K - \boldsymbol{\phi}^0)^T]^T \;, \tag{31}$$

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

Using this formulation, while minimizing the errors $\mathbf{e}$, we penalize the objective function for any deviates between $\boldsymbol{\phi}^k$ and $\boldsymbol{\phi}^0$, since our only available knowledge is that only a few parameters in $\boldsymbol{\phi}^k$ 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 [9,27].

A practical application to FET modeling has been described by Bandler et al. [16] where multiple circuits were created by taking three sets of actual measurements under different biasing conditions.

Another important application of multi-circuit modeling is to create analytical formulas which link the model $\boldsymbol{\phi}$ to the actual physical parameters $\boldsymbol{\phi}^M$. Such formulas will become extremely useful in guiding an actual production alignment or tuning procedure. A sequence of adjustments on $\boldsymbol{\phi}^M$ 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 $\boldsymbol{\phi}^M$ 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.

An example of establishing an experimental relationship between the physical and model parameters for a multi-cavity filter using multiple sets of actual measurements has been described by Daijavad [44].

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. ([12], section V, Test 2).

The commercial packages TOUCHSTONE [104,105] and SUPER-COMPACT [99] 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.

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 [68], Pinel and Roberts [87], Butler [36], Elias [52], Bandler, Liu and Tromp [24]. During the years, significant contributions have been made by, among others, Director and Hachtel [47] (the simplicial method), Soin and Spence [98] (the gravity method), Bandler and Abdel-Malek [1,2,7] (multi-dimensional approximation), Biernacki and Styblinski [30] (dynamic constraint approximation), Polak and Sangiovanni-Vincentelli [90] (a method using outer approximation), as well as Singhal and Pinel [97] (the parametric sampling method). Following the review, we propose a generalized ℓ_p centering algorithm.

A commonly assumed cost versus yield curve [97] is shown in Fig. 9. 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 10 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. [23,24] have formulated it as a nonlinear programming problem as

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

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 [11] have demonstrated a solution to (32) by minimax optimization. Polak and Sangiovanni-Vincentelli [90] 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 $R_a = \{\phi \mid H(e(\phi)) \leq 0\}$.

Director and Hachtel [47] have devised a simplicial approximation approach. It begins by determining points ϕ^k on the boundary of R_a which is given by $\Omega_a = \{\phi \mid H(e(\phi)) = 0\}$. 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 the 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 [1,2,7] 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 [30] 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 [98] 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,

$$\begin{aligned} \boldsymbol{\phi}^{f} &= \left[\sum_{k \in J} \boldsymbol{\phi}^{k} \right] / K_{fail} , \\ \boldsymbol{\phi}^{p} &= \left[\sum_{k \notin J} \boldsymbol{\phi}^{k} \right] / K_{pass} , \end{aligned} \tag{34}$$

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 $\boldsymbol{\phi}^0$ is then adjusted along the direction $\mathbf{s} = \boldsymbol{\phi}^p - \boldsymbol{\phi}^f$ 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 [97] 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

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

where $I_a(\boldsymbol{\phi})$ is the acceptance index defined in (20) and $\Gamma(\boldsymbol{\phi},\mathbf{x})$ 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 $\boldsymbol{\phi}^k$, $k = 1, 2,..., K$, from the component density Γ, perform K circuit analyses and then take the average of $I_a(\boldsymbol{\phi}^k)$. 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 parametric sampling method is based on the concept of importance sampling as

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

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

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

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

This approach has two clear advantages. Firstly, once the indices $I_a(\boldsymbol{\phi}^k)$ 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 $h(\boldsymbol{\phi})$ 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 $I_a(\boldsymbol{\phi}^k)$ is defined as 1 if $\boldsymbol{\phi}^k$ is acceptable after tuning. If $\boldsymbol{\phi}^k$ 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 $\boldsymbol{\phi}^0$ is to be optimized. Define

$$\mathbf{f} = [\mathbf{e}^T(\boldsymbol{\phi}^1) \ldots \mathbf{e}^T(\boldsymbol{\phi}^K)]^T \tag{38}$$

as the set of multi-circuit error functions. We can achieve a worst-case minimax design by

$$\underset{\mathbf{x}}{\text{minimize}}\ U(\mathbf{x}) = H_\infty(\mathbf{f}) = \max_k \max_j \{e_j(\boldsymbol{\phi}^k)\}, \tag{39}$$

where the multiple circuits $\boldsymbol{\phi}^k$ are related to $\boldsymbol{\phi}^0$ 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. We may attempt to use a generalized ℓ_2 or ℓ_1 function (i.e., $H_2(\mathbf{f})$ or $H_1(\mathbf{f})$) instead of $H_\infty(\mathbf{f})$ in (39), hoping to reduce the emphasis given to the worst case.

In order to gain more insight into the problem, we define, for each $\boldsymbol{\phi}^k$, a scalar function which will indicate directly whether $\boldsymbol{\phi}^k$ satisfies or violates the specifications and by how much. For this purpose, we choose a set of generalized ℓ_p functions as

$$v_k(\mathbf{x}) = H_p(\mathbf{e}(\boldsymbol{\phi}^k)), \qquad k = 1, 2, \ldots, K. \tag{40}$$

The sign of v_k indicates the acceptability of $\boldsymbol{\phi}^k$ while the magnitude of v_k measures, so to speak, the distance between $\boldsymbol{\phi}^k$ 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

$$\underset{\mathbf{x}}{\text{minimize}}\ U(\mathbf{x}) = H_p(\mathbf{u}(\mathbf{x})), \tag{41}$$

where

$$\mathbf{u}(\mathbf{x}) = \begin{bmatrix} \alpha_1 v_1 \\ \cdot \\ \cdot \\ \alpha_K v_K \end{bmatrix} = \begin{bmatrix} \alpha_1 H_q(\mathbf{e}(\boldsymbol{\phi}^1)) \\ \cdot \\ \cdot \\ \alpha_K H_q(\mathbf{e}(\boldsymbol{\phi}^K)) \end{bmatrix} \tag{42}$$

and $\alpha_1, \alpha_2, \ldots, \alpha_K$ 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 $U(\mathbf{x}) \le 0$ 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 $U(\mathbf{x}) \le 0$ (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 (39) 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 (41). Also, given a starting point $\mathbf{x}_0$, we define the set of multipliers by

$$\alpha_k = 1/|v_k(\mathbf{x}_0)|, \quad k = 1, 2, \ldots, K \tag{43}$$

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

$$\sum_{k \in J} [u_k(\mathbf{x})]^p \tag{44}$$

where $J = \{k \mid u_k > 0\}$. As $p \to 0$ (44) 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

$$U(\mathbf{x}) = \sum_{k \in J} u_k(\mathbf{x}) = \sum_{k \in J} \alpha_k v_k(\mathbf{x}) . \tag{45}$$

With the multipliers defined by (43), 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.

One important feature of this approach is its capability of accommodating arbitrary tolerance distributions, since they only influence the generation of ϕ^k. The numerical results we have obtained are very promising. The generalized ℓ_p centering algorithm can also be extended to including variable tolerances and tuning.

VI. EXAMPLES OF STATISTICAL DESIGN

Example 1

The classical two-section 10:1 transmission line transformer, originally proposed by Bandler et al. [23] 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. 11 shows the minimax contours, the minimax nominal solution, as well as the worst-case solutions [23] for

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

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

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. Two problems of less than 100% yield have also been considered by Bandler and Abdel-Malek [7] 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. 12. For more details see the original paper [7].

Example 2

The statistical design of a Chebyshev lowpass filter (Singhal and Pinel [97]) is used as the second example. 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.

Singhal and Pinel [97] have applied the parametric sampling method to the same circuit assuming normal distributions for the toleranced elements. But, as we have pointed out earlier in this paper, the parametric sampling method can not be applied to non-differentiable (such as uniform) distributions. Here, we consider 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 as given in [97] 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 I.

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.

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 $\mathbf{x}_k$.

We use the notation $U(\mathbf{x})$ 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 individual error functions so that U = H(f). We also use f_j' for the first-order derivatives of f_j and G for the Jacobian matrix of f.

ℓ_p Optimization and Mathematical Programming

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

$$\underset{\mathbf{x},\,\mathbf{y}}{\text{minimize}} \quad \sum_{j=1}^{m} y_j \tag{46}$$

subject to

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

Other equivalent formulations are summarized in Table II. For the convenience of presentation, we denote these mathematical programming problems by P(**x**,**f**). One important feature of P(**x**,**f**) is that it has a linear or quadratic objective function. If **f** is a set of linear functions, then P(**x**,**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(**x**,**f**,D) be the problem of P(**x**,**f**) subject to a set of linear constraints of the form

$$\text{D:} \quad \begin{array}{ll} \mathbf{a}_\ell^T \mathbf{x} + b_\ell = 0\,, & \ell = 1, 2, \ldots, L_{eq}\,, \\ \mathbf{a}_\ell^T \mathbf{x} + b_\ell \geq 0 & \ell = L_{eq} + 1, \ldots, L\,, \end{array} \tag{47}$$

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

$$\bar{\mathbf{f}}(\mathbf{h}) = \mathbf{f}(\mathbf{x}_k) + \mathbf{G}(\mathbf{x}_k)\,\mathbf{h}\,, \tag{48}$$

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

$$D:\quad \begin{aligned} \Lambda_k &\geq h_j\,, & j &= 1, 2, ..., n\,,\\ \Lambda_k &\geq -h_j\,, & j &= 1, 2, ..., n\,. \end{aligned} \tag{49}$$

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 [78]) to $U(\mathbf{x}) = H(\mathbf{f})$ resulting in

$$\bar{U}(\mathbf{h}) = H(\bar{\mathbf{f}}(\mathbf{h}))\,. \tag{50}$$

It is important to point out that (50) 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. At each iteration, the local bound Λ_k in (49) is adjusted according to the goodness of the linearized model.

The above describes the essence of a class of algorithms due to Madsen who has called it Method 1. Madsen [78] has shown that the algorithm provides global convergence in which the proper use of trust regions constitutes a critical part. Such a method has been implemented as an important element in the minimax and ℓ_1 algorithm of Hald and Madsen [65,66]. In some other earlier work by Osborne and Watson [85,86] 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 [76,81] method. Given $\mathbf{x}_k$, it solves

$$\underset{\mathbf{h}}{\text{minimize}}\ \mathbf{h}^T(\mathbf{G}^T\mathbf{G} + \theta_k 1)\mathbf{h} + 2\mathbf{f}^T\mathbf{G}\mathbf{h} + \mathbf{f}^T\mathbf{f}\ , \tag{51}$$

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

$$(\mathbf{G}^T\mathbf{G} + \theta_k 1)\,\mathbf{h}_k = \mathbf{G}^T\mathbf{f} \tag{52}$$

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 (52). Furthermore, it plays, roughly speaking, an inversed role of Λ_k to control the size of a trust region. When $\theta_k \rightarrow \infty$, $\mathbf{h}_k$ 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 $\Lambda_k \rightarrow \infty$.

The concept of trust region has been discussed in a broader context by Moré in a recent survey [82].

Quasi-Newton Methods

Quasi-Newton methods (also known as variable metric methods) are originated in and steadily upgraded from the work of Davidon [45], Broyden [33,34] as well as Fletcher and Powell [55].

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

$$\mathbf{h}_k = -\alpha_k\,\mathbf{B}_k^{-1}\,\nabla U(\mathbf{x}_k)\,, \tag{53}$$

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 [70] to the equivalent problem $P(\mathbf{x},\mathbf{f})$, we shall find a set of optimality equations

$$\mathbf{R}(\mathbf{x}) = \mathbf{0}\,. \tag{54}$$

Since a local optimum $\mathbf{x}^*$ must satisfy these equations, we are naturally motivated to solve (54), as a means of finding the minimizer of $U(\mathbf{x})$. A quasi-Newton step for solving nonlinear equations (54) is given by

$$\mathbf{h}_k = -\alpha_k \mathbf{J}_k^{-1} \mathbf{R}(\mathbf{x}_k)\,, \tag{55}$$

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 (55) reverts to (53).

Hald and Madsen [65,66] and Bandler et al. [21,22] have described the implementation of a quasi-Newton method for the minimax and ℓ_1 optimization in which the objective functions are not differentiable. Clarke [43] 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 (53) or (55), 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 [91], the Davidon-Fletcher-Powell (DFP) update [45,55] and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update [35,53,60,95]. The merits of these formulas and a great many other variations are often compared in terms of their preservation of positive definiteness, convergence to the true Hessian and numerical performance (see for instance Fletcher [54], Gill and Murray [59]).

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 [50] has shown that theoretically all members of the Broyden family [34,53] 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.

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 [65,66,78] 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 [21,22,79,80].

Powell [92] 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 [56,75,88] provide an alternative for such problems. 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.

VIII. GRADIENT CALCULATION AND APPROXIMATION

The application of gradient-based ℓ_p optimization methods requires the first-order derivatives of the error functions with respect to the variables.

In circuit optimization, these derivatives are usually obtained from a sensitivity analysis of the network under consideration. For linear circuits in the frequency domain, it is often possible to calculate the exact sensitivies by the adjoint network approach [5,31,48].

However, 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 [99] and TOUCHSTONE [104,105] 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 [51], Gustavson [61], Hachtel et al. [62]). Another possibility is to rearrange the overall nodal matrix into a bordered block structure which is then solved using the Sherman-Morrison-Woodbury formula [63,96]. Sometimes it is also possible to develop efficient formulas for a special structure, such as the approach of Bandler et al. [17] for branched cascaded networks.

In practice, 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.

In cases where either exact sensitivities do not exist or they are too difficult to calculate, we can utilize gradient approximations [15,16,77,109]. A recent approach to circuit optimization with integrated gradient approximations has been described by Bandler et al. [16]. It has been shown to be very effective and efficient in practical applications including FET modeling amd multiplexer optimization.

IX. 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.

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 [57, 58], Haimes [64], Lasdon [72]), third generation simulation techniques (Hachtel and Sangiovanni-Vincentelli [63]), fault diagnosis (Bandler and Salama [27]), supercomputer-aided CAD (Rizzoli et al. [93]), the simulated annealing and combinatorial optimization methods and their application to integrated circuit layout problems [38,69,84], and the new automated decomposition approach to large scale optimization (Bandler and Zhang [28]).

This paper is particularly timely as software, based on techniques which we have described, is being integrated by Optimization Systems Associates Inc. into SUPER-COMPACT by arrangement with Compact Software Inc.

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 useful comments offered by the reviewers are also appreciated. 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. Thanks are extended to Dr. R.A. Pucel of Raytheon Company, Research Division, Lexington, MA, for reviving the first author's interest and work in design centering and yield optimization. Dr. U.L. Rohde of Compact Software Inc., Paterson, NJ, is facilitating the practical implementation of advanced mathematical techniques of CAD. The stimulating environment provided by Drs. Pucel and Rohde to the first author is greatly appreciated.

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

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

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

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

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] J.W. Bandler, S.H. Chen, S. Daijavad and K. Madsen, "Efficient optimization with integrated gradient approximations", IEEE Trans. Microwave Theory Tech., vol. MTT-36, 1988.

[17] 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.

[18] 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.

[19] 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.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] 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.

[25] 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.

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

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

[28] 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, pp. 709-712.

[29] 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.

[30] 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.

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

[32] 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.

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

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

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

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

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

[38] A. Casotto, F. Romeo and A.L. Sangiovanni-Vincentelli, "A parallel simulated annealing algorithm for the placement of macro-cells", Proc. IEEE Int. Conf. on Computer-Aided Design (Santa Clara, CA), 1986, pp. 30-33.

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

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

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

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

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

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

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

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

[47] 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.

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

[49] 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.

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

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

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

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

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

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

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

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

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

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

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

[61] 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.

[62] 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.

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

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

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

[66] 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.

[67] 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.

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

[69] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi, "Optimization by simulated annealing", Science, vol. 220, 1983, pp. 671-680.

[70] 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.

[71] 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.

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

[73] 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.

[74] 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.

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

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

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

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

[79] 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.

[80] 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.

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

[82] 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.

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

[84] S. Nahar, S. Sahni and E. Shragowitz, "Simulated annealing and combinatorial optimization", Proc. 23rd Design Automation Conf. (Las Vegas, NA), 1986, pp. 293-299.

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

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

[87] 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.

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

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

[90] E. Polak and A.L. 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.

[91] 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.

[92] 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.

[93] 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.

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

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

[96] 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.

[97] 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.

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

[99] SUPER-COMPACT User's Manual, Compact Software Inc., Paterson, NJ 07504, May 1986.

[100] 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.

[101] 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.

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

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

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

[105] TOUCHSTONE Reference Manual (Version 1.5), EEsof Inc., Westlake Village, CA 91362, March 1987.

[106] 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.

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

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

[109] 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 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.

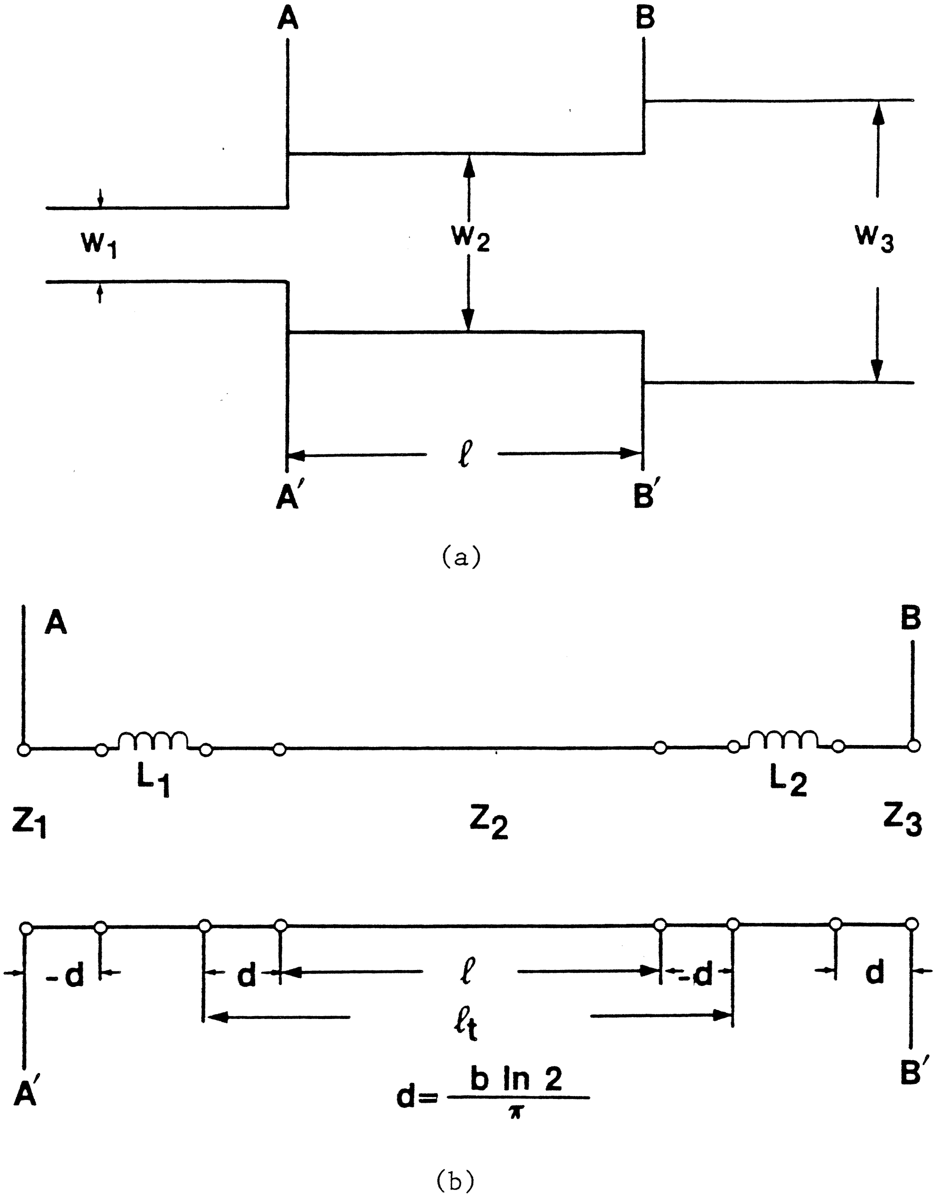
TABLE II

MATHEMATICAL PROGRAMMING EQUIVALENT FORMULATIONS FOR ℓ_1, ℓ_2 and ℓ_∞ OPTIMIZATION

The original problem:	$\underset{\mathbf{x}}{\text{minimize}}\ H(\mathbf{f})$
The equivalent problem:	$\underset{\mathbf{x},\mathbf{y}}{\text{minimize}}\ V(\mathbf{x},\mathbf{y})$ 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 $p < \infty$.



(a)

(b)

Fig. 1 A microwave stripline transformer showing (a) the physical structure and (b) the equivalent circuit model [25]. 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 [25].

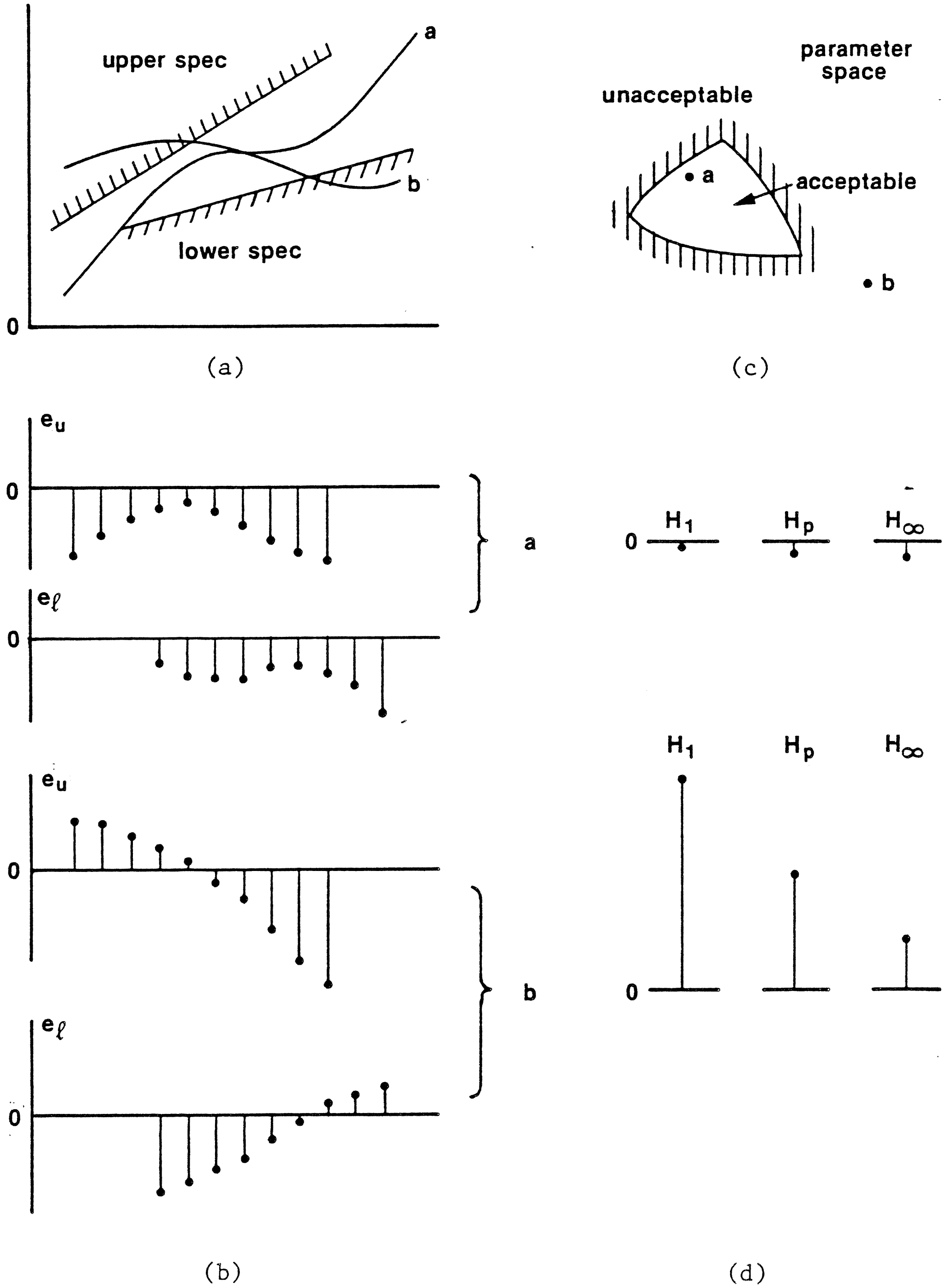


Fig. 2 Illustrations of (a) upper specifications, lower specifications and the responses of circuits a and b, (b) error functions corresponding to circuits a and b, (c) the acceptable region and (d) generalized ℓ_p objective functions defined in (13).

f

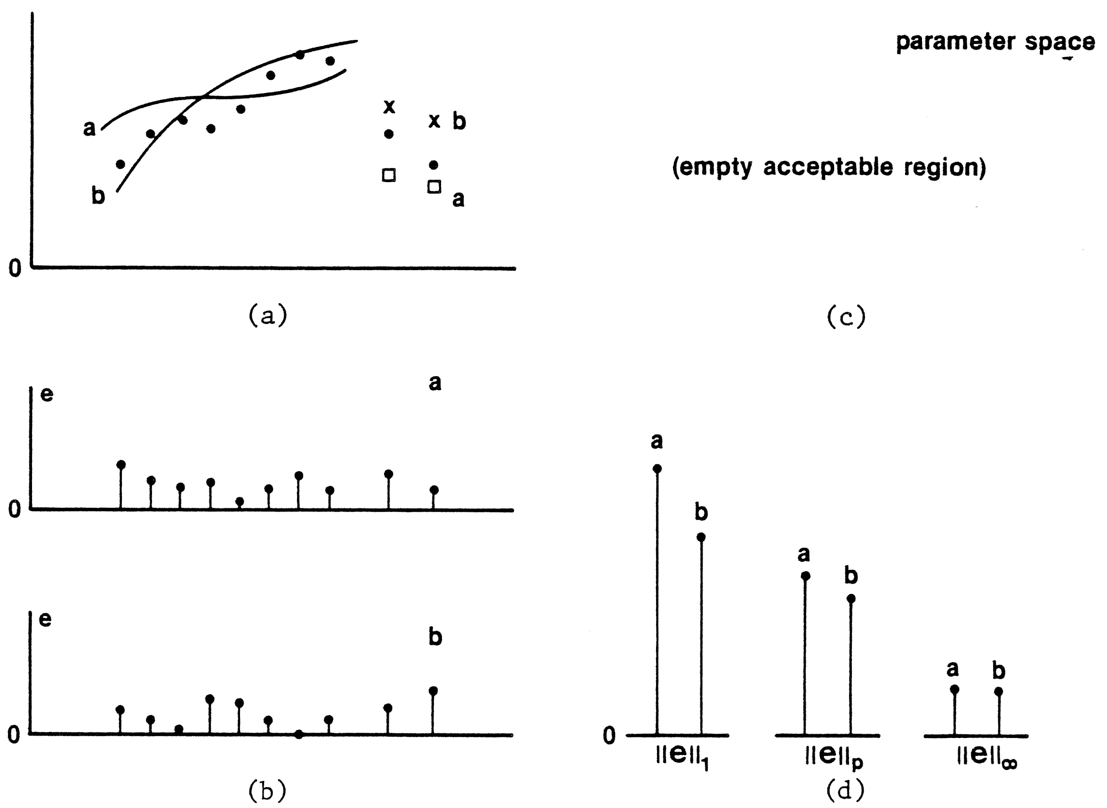


Fig. 3 Illustrations of (a) a discretized single specification and two discrete single specifications (e.g., expected parameter values to be matched), as well as the responses of circuits a and b, (b) error functions related to circuits a and b, (c) the (empty) acceptable region (i.e., a perfect match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

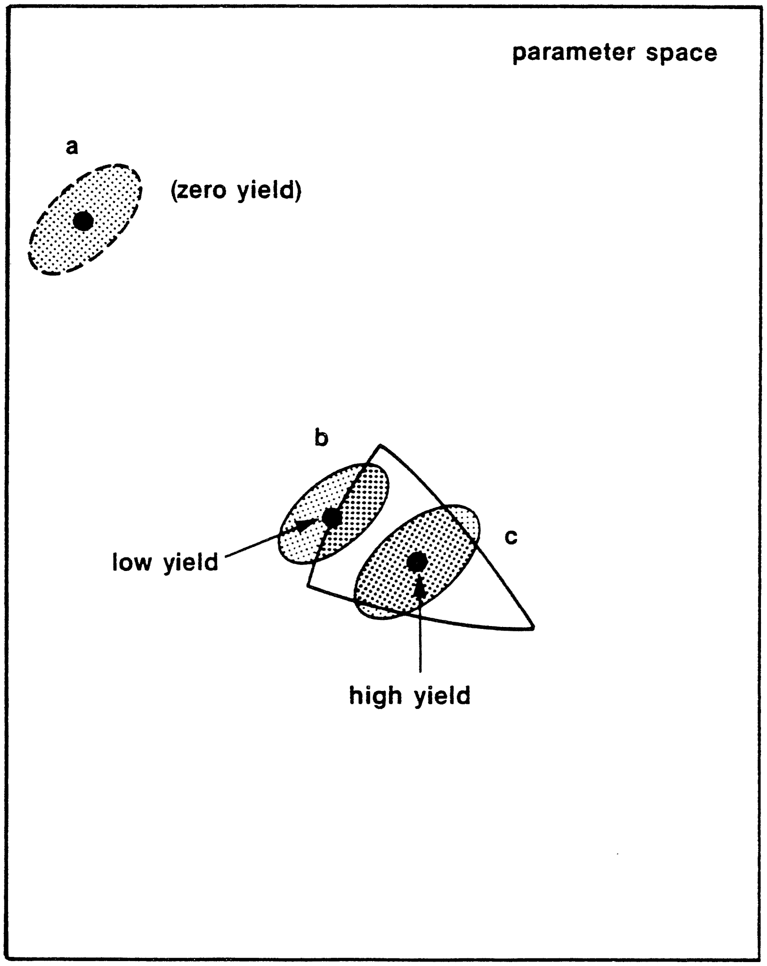


Fig. 4 Three nominal points and the related yield.

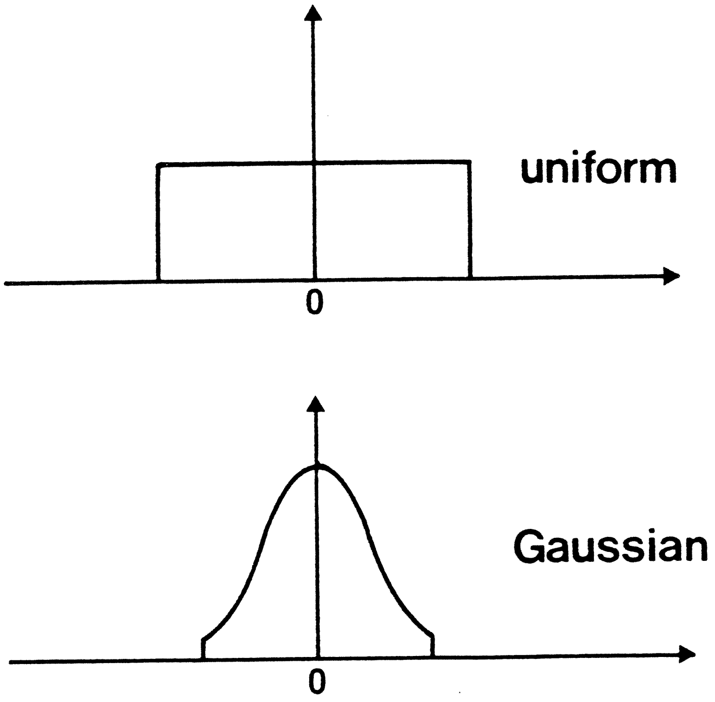


Fig. 5 Typical tolerance distributions: uniform and Gaussian (normal).

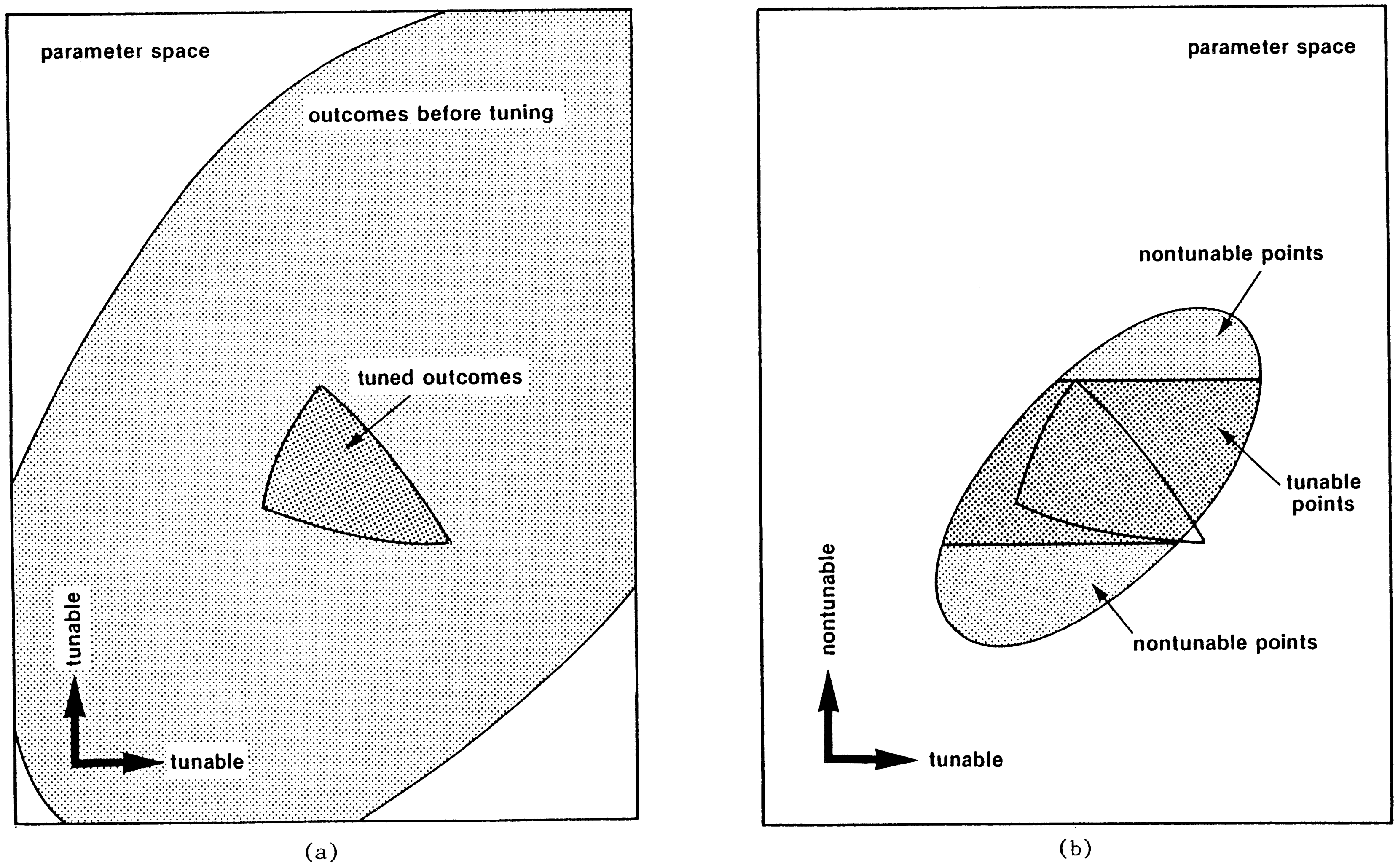
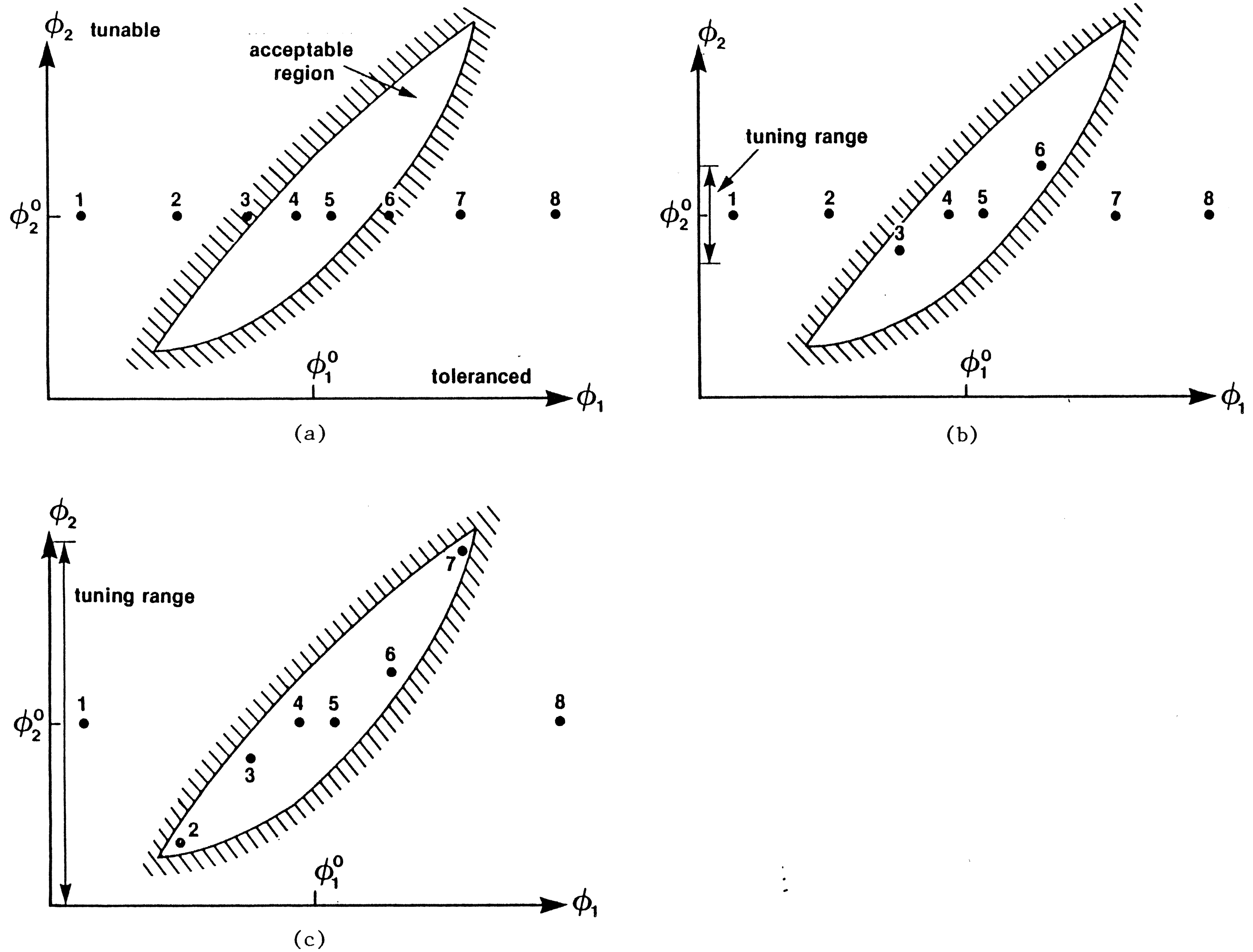


Fig. 6 Illustrations of tuning: (a) both parameters are tunable for a case in which the probability that an untuned design meets the specifications is very low and (b) only one parameter is tunable.



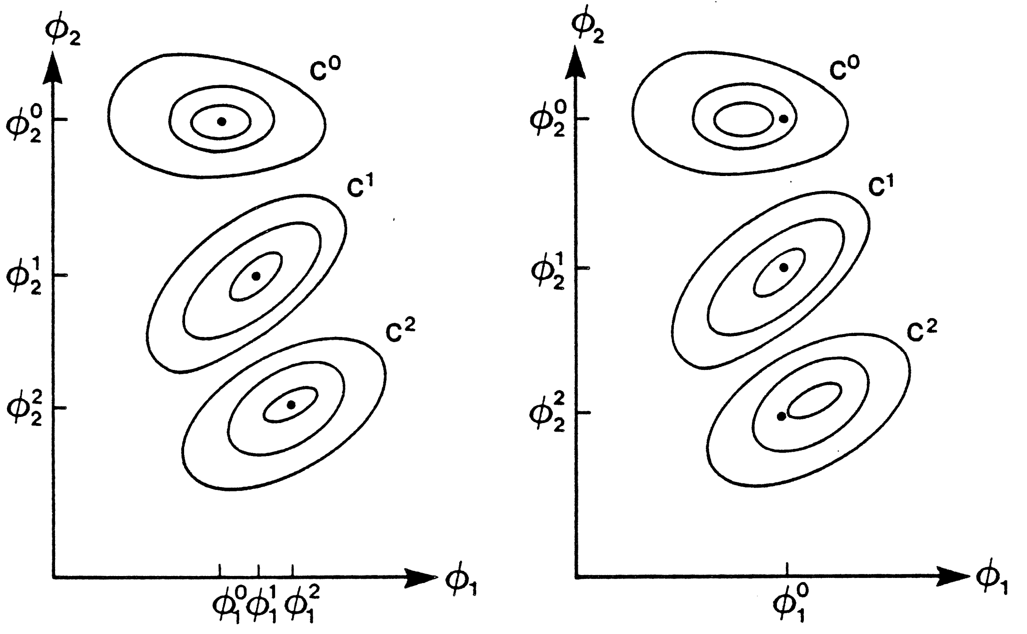
(a)

(b)

(c)

Fig. 7 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.



(a) (b)

Fig. 8 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 $\boldsymbol{\phi}^0$, $\boldsymbol{\phi}^1$ and $\boldsymbol{\phi}^2$. ${\phi_1}^0$, ${\phi_1}^1$ 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.

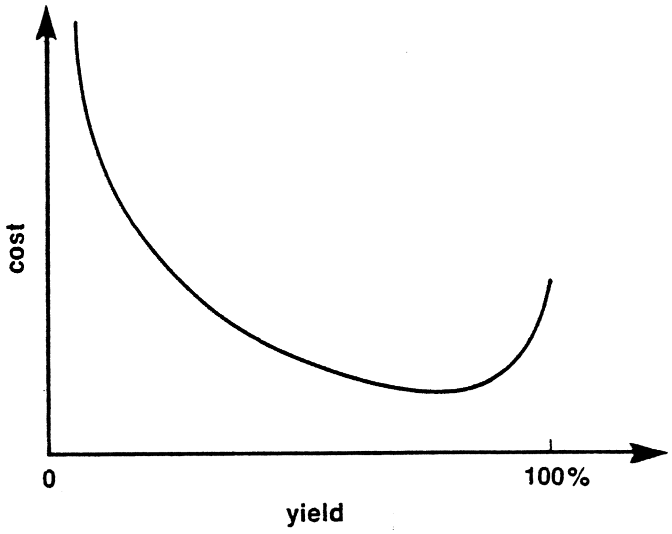


Fig. 9 A typical cost-versus-yield curve [97].

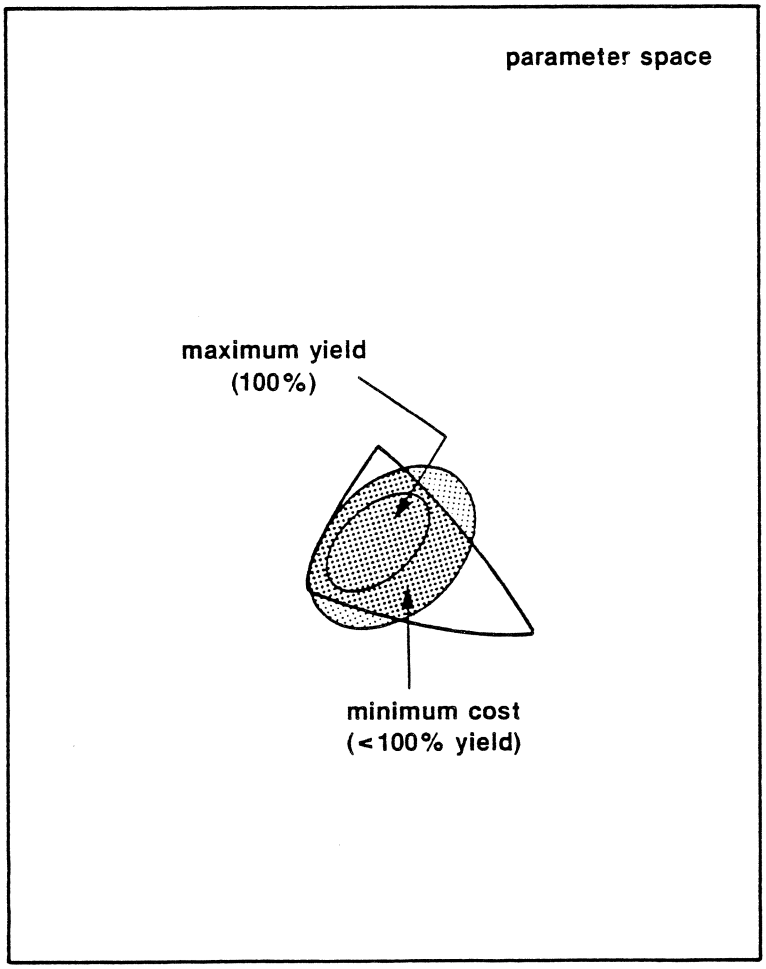


Fig. 10 A maximum yield design and a minimum cost design.

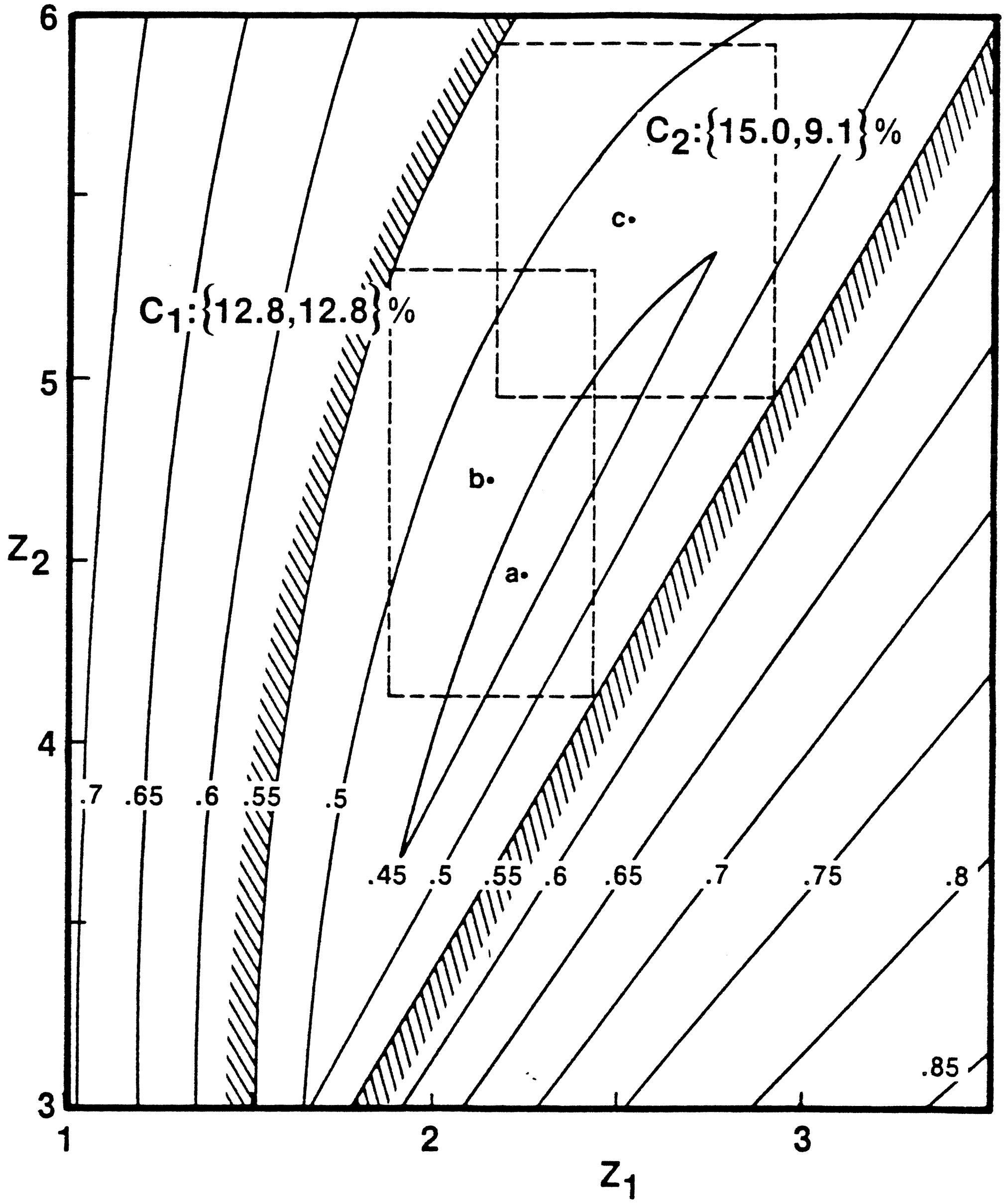


Fig. 11 Contours of max $|\rho_j|$ with respect to Z_1 and Z_2 for the two-section transformer indicating the minimax nominal solution a, the centered design with relative tolerances b and the centered design with absolute tolerances c. The values in brackets are the optimized tolerances (as percentages of the nominal values). The specification is $|\rho| \leq 0.55$.

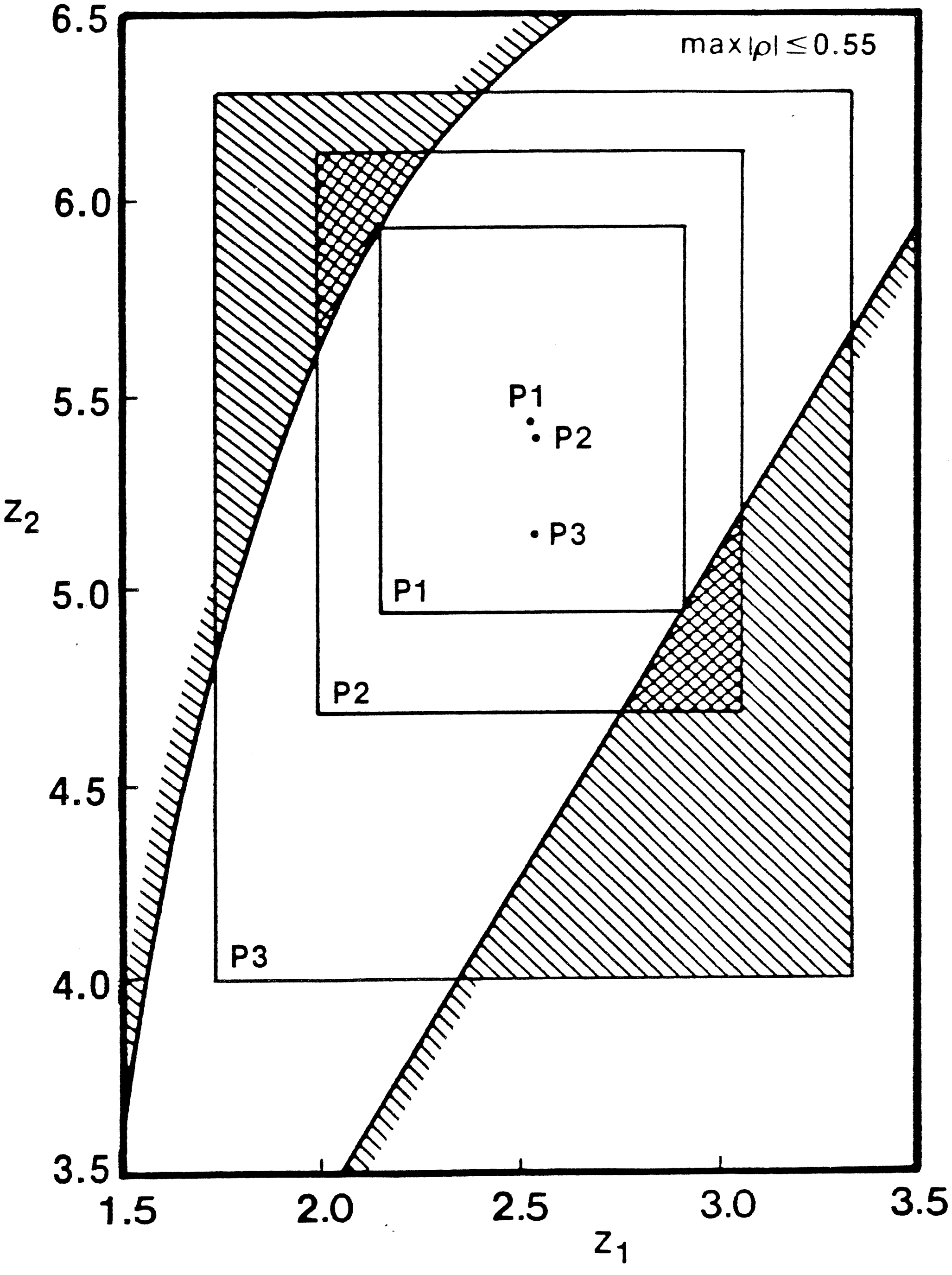


Fig. 12 The optimized tolerance regions and nominal values for the worst case design P1, 90% yield design P2 and minimum cost design P3 of the two-section transformer.

AN AUTOMATIC DECOMPOSITION APPROACH TO OPTIMIZATION OF LARGE MICROWAVE SYSTEMS

John W. Bandler, Fellow, IEEE, and Qi-jun Zhang, Student Member, IEEE

Abstract

We present a novel and general technique applicable to the optimization of large microwave systems. Using sensitivity information obtained from a suitable Monte-Carlo analysis, we extract possible decomposition properties which could otherwise be deduced only through a detailed physical and topological investigation. The overall problem is automatically separated into a sequence of subproblems, each being characterized by the optimization of a subset of circuit functions w.r.t. variables which are sensitive to the selected responses. A heuristic algorithm for automatic decomposition is developed. The decomposition patterns are dynamically updated until a satisfactory solution is reached. The partitioning approach proposed by Kondoh for FET modelling problems is verified The technique was successfully tested on large scale optimizations of microwave multiplexers involving 16 channels, 399 nonlinear functions and 240 variables.

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under Grants A7239 and G1135 and in part by Optimization Systems Associates, Inc.

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

A serious challenge to researchers in microwave CAD areas is due to the size of practical microwave systems. Existing CAD techniques, mature enough to handle systems of ordinary size, generally balk at large circuits. The reasons for their failure include prohibitive computer storage and CPU times required. A frequent frustration with large scale optimization is the increased likelihood of stoppping at an undesired local optimum. Other difficulties, especially in prototype and production tuning, are due to human inability to cope with problems involving large numbers of independent variables to be adjusted simultaneously to meet a specified response pattern over a wide frequency range.

Recently, FET modelling [1] and manifold multiplexer design [2] problems were solved using appropriate decomposition schemes. The success of these efforts motivated us to pursue the generalization and automation of decomposition approaches for microwave optimization problems.

The concept of decomposition has been a traditional mathematically based vehicle for approaching large scale problems, e.g., in mathematical programming [3-6], in circuit analysis [7-13], design [3] and fault diagnosis [14] and in optimal power flow [15,16], state estimation[17] and real and reactive power optimization problems [18].

Microwave engineers have their own special concerns with decomposition, as exposed in [1, 2, 19]. Thorough laboratory experimentation has to be performed before using certain function structures assumed in mathematical programming theory. They do not take advantage of topological analysis often exploited in the areas of circuits and systems since microwave device models are oriented more to physical than topological analysis. Unlike power systems, most microwave responses are much more complicated and highly nonlinear. It is often difficult for microwave engineers to analytically indicate possible decomposition patterns. To our knowledge, there does not exist a general and abstract theory describing a decomposition approach to microwave circuit optimization not requiring particular physical or topological knowledge of the system.

In this paper, we present a novel technique applicable to the optimization of large microwave systems. Using sensitivity information obtained from a suitable Monte-Carlo analysis, we extract possible decomposition properties which could otherwise be deduced only through a physical and topological investigation. The overall problem is automatically separated into a sequence of subproblems, each being characterized by the optimization of a subset of circuit functions w.r.t. variables which are sensitive to the selected responses. Our suggested technique has been successfully tested on microwave multiplexers involving up to 16 channels and 240 variables.

In Section II, we describe the basic concepts of decomposition for circuit optimization problems. Using these concepts, the partitioning approach for FET modelling problems suggested by Kondoh[1] is verified. Section III illustrates the automatic determination of suboptimiztion problems. An automated decomposition algorithm for large scale microwave optimization is presented in Section IV. In Section V, the method is applied to the optimization of microwave multiplexers. Interesting results demonstrating the procedure of automated decomposition for a 5-channel multiplexer are depicted in illustrative graphs. The results of optimizing a 16-channel multiplexer using our approach are provided.

II. THE DECOMPOSITION APPROACH

Circuit Optimization Problems

Let $\boldsymbol{\phi} = [\phi_1 \;\; \phi_2 \;\; \ldots \;\; \phi_n]^T$ represent the system parameters. The circuit responses, denoted as $F_k(\boldsymbol{\phi}, \omega)$, $k = 1, 2, \ldots, n_F$, are functions of variables $\boldsymbol{\phi}$ and frequency ω. In an optimization problem for circuit design, the objective function usually involves a set of nonlinear error functions $f_j(\boldsymbol{\phi})$, $j = 1, 2, \ldots, m$. Typically, the error functions represent the weighted differences between circuit responses and given specifications in the form

$$\begin{gathered} w_{Uk}(\omega)(F_k(\boldsymbol{\phi}, \omega) - S_{Uk}(\omega)) \\ - w_{Lk}(\omega)(F_k(\boldsymbol{\phi}, \omega) - S_{Lk}(\omega)) \qquad (1) \\ k \in \{1, 2, \ldots, n_F\}, \end{gathered}$$

where S_{Uk} and S_{Lk} are upper and lower specifications, respectively. w_{Uk} and w_{Lk} are weighting factors.

Suppose sets I and J are defined as

$$I \triangleq \{1, 2, ..., n\}, \tag{2}$$

$$J \triangleq \{1, 2, ..., m\}. \tag{3}$$

The overall optimization problem, e.g., a minimax optimization, is

$$\underset{\phi_i,\ i \in I}{\text{minimize}} \quad \underset{j \in J}{\max} \quad f_j(\boldsymbol{\phi}). \tag{4}$$

Description of the Decomposition Approach

In a decomposition approach, one attempts to reach the overall solution by solving a sequence of subproblems. A typical subproblem is characterized by

$$\underset{\phi_i,\ i \in I^s}{\text{minimize}} \quad \underset{j \in J^s}{\max} \quad f_j(\boldsymbol{\phi}), \tag{5}$$

where I^s and J^s are subsets of I and J, respectively.

The basic idea for decomposition is to decouple a variable ϕ_i from a function f_j if the interaction between them is weak. A subproblem contains only the sensitively related variables and functions. A proper arrangement of the sequence of different subproblems to be solved is often important to ensure convergence and efficiency.

Sensitivity Analysis

We perform sensitivity analysis at a set of randomly chosen points $\boldsymbol{\phi}^\ell$, $\ell = 1, 2, ...$ A measure of the interaction between ϕ_i and f_j is defined as

$$S_{ij} \triangleq \sum_\ell \left(\frac{\partial f_j(\boldsymbol{\phi}^\ell)}{\partial \phi_i} \frac{\phi_i^0}{f_j^0} \right)^2 , \tag{6}$$

where $\phi_i{}^0$ and $f_j{}^0$ are used for scaling. All the S_{ij}, $i = 1, 2, ..., n$ and $j = 1, 2, ..., m$, constitute a nxm sensitivity matrix **S**. It is reasonable to conclude that ϕ_i and f_j can be decoupled if S_{ij} is very small.

Grouping of Variables and Functions

The examination of various interaction patterns between ϕ_i, $i \in I$, and f_j, $j \in J$, results in the breakdown of all variables $\boldsymbol{\phi}$ into p groups identified by index sets $I_1, I_2, ..., I_p$, and all functions **f** into q groups identified by sets $J_1, J_2, ..., J_q$. We have

$$I = I_1 \cup I_2 \cup ... \cup I_p \tag{7}$$

and

$$J = J_1 \cup J_2 \cup ... \cup J_q \ . \tag{8}$$

The partitioning of $\boldsymbol{\phi}$ or **f** can be achieved either manually or automatically. The manual procedure corresponds to the manual determination of variable groups and function groups using a priori knowledge. Such knowledge is typically obtained through extensive laboratory experiment and an excellent understanding of the particular device. The automatic procedure corresponds to the computerized partitioning of $\boldsymbol{\phi}$ or **f** based upon the sensitivity matrix **S**.

As an example for manual partitioning of **f**, we consider a N-channel multiplexer. The common port return loss and channel insertion loss responses associated with the same channel can be grouped together since their behavior is similarly affected by variables $\boldsymbol{\phi}$. Therefore, we have N groups of functions, i.e., $q = N$. J_ℓ contains indices of error functions related to channel ℓ, $\ell = 1, 2, ..., N$.

Automatic Partitioning of Variables $\boldsymbol{\phi}$

Suppose the function groups have been determined, i.e., J has been decomposed into J_ℓ, $\ell = 1, 2, ..., q$. We define a nx$q$ matrix C whose (i, ℓ)th component is

$$C_{i\ell} \triangleq \sum_{j \in J_\ell} S_{ij} \, . \tag{9}$$

A very small value of an entry in the C matrix, say, $C_{i\ell}$, implies that the ith variable and the ℓth function group are weakly interconnected.

Let C_{ave} represent the average value of all components in the C matrix. For a given factor λ, $\lambda \geq 0$, the matrix is made sparse such that $C_{i\ell}$ is set to zero if it is less than λC_{ave}. By making C sparse, insensitive variables are eliminated and weak interactions between variables and function groups are decoupled.

Two variables ϕ_i and ϕ_j belong to the same group if they interact only with the same groups of functions, i.e., if the ith and the jth rows of C have the same zero/nonzero pattern. A thorough computerized checking of the C matrix results in the automatic determination of index sets I_k, k = 1, 2, ..., *p*.

Example of Matrix C

Consider the fictitious relations between variables and function groups shown in Fig. 1(a). The functions f have been arranged into 5 groups. The C matrix (already made sparse) is

$$\begin{bmatrix} 22. & 100. & 32. & 0. & 0. \\ 0. & 100. & 0. & 0. & 0. \\ 0. & 100. & 0. & 0. & 0. \\ 0. & 0. & 83. & 100. & 0. \\ 0. & 0. & 0. & 0. & 100. \\ 0. & 0. & 100. & 86. & 0. \\ 0. & 0. & 100. & 0. & 0. \\ 0. & 78. & 100. & 55. & 0. \\ 100. & 0. & 0. & 0. & 0. \end{bmatrix} . \qquad (10)$$

As seen from Fig. 1(a), ϕ_2 and ϕ_3 both affect only the 2nd function group. In the C matrix, rows 2 and 3 both have only one nonzero located at the 2nd column. Therefore, variables ϕ_2 and ϕ_3 are grouped together. Similarly, variables ϕ_4 and ϕ_6 belong to the same group. The resulting index sets for variable groups are $I_1 = \{9\}$, $I_2 = \{2, 3\}$, $I_3 = \{7\}$, $I_4 = \{5\}$, $I_5 = \{4, 6\}$, $I_6 = \{1\}$ and $I_7 = \{8\}$. The index sets have been ordered such that the kth variable group correlates with no more function groups than the (k+1)th variable group does, k = 1, 2, ..., 6. Such an arrangement is made to keep subsequent description simple.

Decomposition Dictionary

To manipulate directly with groups of variables and groups of functions, we construct a *pxq* dictionary decomposition matrix **D**. Define the (k, ℓ)th component of **D** as

$$\begin{aligned} D_{k\ell} &\triangleq \sum_{i\in I_k} \sum_{j\in J_\ell} S_{ij} \\ &= \sum_{i\in I_k} C_{i\ell} \,. \end{aligned} \tag{11}$$

If $D_{k\ell}$ is zero, variables in the kth group are decoupled from functions in the ℓth group. Otherwise if $D_{k\ell} \neq 0$, we say that ϕ_i, $i\in I_k$, and f_j, $j\in J_\ell$, are correlated. The decomposition dictionary gives a clear picture of the correlation patterns between groups of variables and functions, facilitating the automatic determination of suboptimization problems. The ideal dictionary is a diagonal matrix where a subproblem simply corresponds to a diagonal element. In this case, only one variable group and one function group is involved in a subproblem. If a diagonal dictionary can be obtained without artificially making **C** sparse (i.e., using sparse factor $\lambda = 0$), then the system is completely decomposable[20]. For a completely decomposable system, different subproblems can be calculated in parallel.

Example of the Decomposition Dictionary

Consider the previous example with the resulting **C** matrix defined in (10). According to the index sets I_k, k = 1, 2, ..., 7, the decomposition dictionary **D** can be obtained from **C** by adding rows 2 and 3, and adding rows 4 and 6, respectively. The relations between groups of variables and functions are shown in Fig. 1(b). The resulting dictionary is

$$\begin{bmatrix} 100. & 0. & 0. & 0. & 0. \\ 0. & 200. & 0. & 0. & 0. \\ 0. & 0. & 100. & 0. & 0. \\ 0. & 0. & 0. & 0. & 100. \\ 0. & 0. & 180. & 180. & 0. \\ 20. & 100. & 30. & 0. & 0. \\ 0. & 70. & 100. & 50. & 0. \end{bmatrix} , \qquad (12)$$

where each entry has been rounded to multiples of 10.

Decomposition for FET Device Models

Through extensive experiment on practical FET devices, Kondoh[1] summarized 8 suboptimization problems which can be repeatedly solved to yield a FET model with improved accuracy. The equivalent circuit is shown in Fig. 2. We perform sensitivity analysis at 10 randomly chosen parameter points in the 10% neighborhood of ϕ^0, ϕ^0 representing the true value listed in [1]. The function f_j used in (6) is defined as the weighted difference between the calculated and the measured values of the modulus or the phase of a particular S parameter. The entire frequency band for calculating S is [1.5, 26.5] GHz. Functions associated with the same S parameter are grouped together. Table I shows the **C** matrix of (9) before being made sparse, indicating strong as well as weak interconnections between each individual parameter and different groups of functions. In the table, each row has been scaled. Table II provides an example of the decomposition dictionary calculated and normalized from Table I. Table II yields 8 subproblems which agree with and further verify the decomposition scheme proposed in [1]. When the **C** matrix is made sparse, certain entries, whose values are only slightly less than the dominant ones, are also set to zero. Therefore, as mentioned in [1], repeated cycling and careful ordering of the 8 suboptimizations are necessary. The feasibility of computerized automatic decomposition is demonstrated by this example.

III. AUTOMATIC DETERMINATION OF SUBOPTIMIZATION PROBLEMS

Reference Function Group and Candidate Variable Group

Usually, the decomposition dictionary is not diagonal. A suboptimization often involves several function groups and several variable groups. Among the function groups involved, there is a key group which we call the reference group. Such a group typically contains the worst error function. The reference function group is used to initiate a subproblem as described in the subsequent text.

Suppose the index set J_ℓ indicates the reference function group. The candidate groups of variables to be used for the suboptimization are those which affect $f_j, j \in J_\ell$.

In the decomposition dictionary, the ℓth column associates with the reference function group. Rows having a nonzero in the ℓth column are candidate rows, each corresponding to a candidate variable group. Take Fig. 1(b) as an example. Suppose that the function group J_2 is the reference group, i.e., $\ell = 2$. The candidate groups of variables are I_2, I_6 and I_7 since they correlate with the reference function group. Correspondingly, in the **D** matrix of (12), rows 2, 6 and 7 are candidate rows since they all have a nonzero in the 2nd column.

Determination of a Suboptimization Problem

An automatic procedure for the determination of I^s and J^s for the suboptimzation of (5) has been developed. Suppose J_ℓ indicates the reference function group. For a selected candidate variable group, e.g., the one corresponding to set I_k, the index set J^s indicates the union of all function groups which correlate with variable group k. I^s identifies variables in the kth group, as well as all other variables which correlate with functions only within $f_j, j \in J^s$. Also, I^s excludes variables not correlating with any active functions in $f_j, j \in J^s$. A function f is said to be active if

$$\begin{aligned} f &> 0.8M_f \quad \text{when} \quad M_f > 0 \\ f &> 1.25M_f \quad \text{when} \quad M_f < 0\,, \end{aligned} \tag{13}$$

where

$$M_f \triangleq \max_{j \in J^s} f_j . \tag{14}$$

Priority of Candidate Variable Groups

It can be seen that a pair of (I^s, J^s) associate with a pair of (I_k, J_ℓ). For a selected reference function group, each candidate variable group leads to a subproblem. The sequence of subproblems used to penalize f_j, $j \in J_\ell$, is determined by the priority of all resulting candidates.

Since each candidate determines the function set J^s for a suboptimization, the priority of the candidate is based upon the pattern of error functions it will affect, i.e. patterns of f_j, $j \in J^s$. Firstly, the fewer the number of function groups in J^s, the higher the priority. Secondly, the worse the overall error functions in J^s, the higher the priority. The overall error functions in J^s are ranked by an appropriate measure, e.g., the generalized least pth function (GLP)[21].

The priority of candidate variable groups can be similarly determined in the decomposition dictionary. The fewer the number of nonzeros that exist in a candidate row, the higher the priority. For two candidate rows containing an equal number of nonzeros, a higher priority is given to the candidate having a larger value in its generalized least pth function.

Example

For the example of Fig. 1, suppose that the maximum error functions within each of the 5 function groups are [3.8 4. 1. −1. 2.]. Suppose that we choose the worst group, i.e., group 2, as the reference function group. According to our previous discussions, the candidate variable groups are I_2, I_6 and I_7. I_2 has the highest priority since it affects fewer (i.e. only one) function groups than I_6 or I_7 does. The functions affected by variables in I_6 (or I_7) are f_j,

$j \in J^s = J_1 \cup J_2 \cup J_3$ (or $J^s = J_2 \cup J_3 \cup J_4$). I_6 has a higher priority than I_7 since the overall error functions in $J_1 \cup J_2 \cup J_3$ are worse than that in $J_2 \cup J_3 \cup J_4$.

Correspondingly, in the decomposition dictionary of (12), rows 2, 6 and 7 are candidates. Row 2 has the highest priority since it contains fewer nonzeros than others. Row 6 has the second highest priority since its GLP value is larger than the GLP value for row 7.

To formulate a suboptimization problem, i.e., to decide I^s and J^s, we choose a pair of (I_k, J_ℓ), e.g., candidate variable group I_6 and reference function group J_2. The index set $J^s = J_1 \cup J_2 \cup J_3$. The variable index set I^s includes I_6 (indicating the candidate variable group), as well as I_1, I_2 and I_3 (indicating all other variables affecting functions only within J^s). Further, I_3 can be excluded from I^s since variables in I_3 do not affect active functions in J^s. Therefore, we have $I^s = I_6 \cup I_1 \cup I_2$.

Circuit Responses and Sample Frequencies

When a subset of error functions $f_j(\boldsymbol{\phi})$, $j \in J^s$, are included in a subproblem, the necessary circuit response functions $F_a(\boldsymbol{\phi}, \omega_b)$, $a \in \{1, 2, ..., n_F\}$ and frequency points ω_b, $b \in \{1, 2, ..., n_\omega\}$, should be selected for circuit simulation programs. This is accomplished using a coding scheme representing the one-to-one correspondence between j and (a, b). We define weighting factor matrices $\mathbf{W}_U$ (for upper specification) and $\mathbf{W}_L$ (for lower specification). Both matrices are n_F by n_ω. The (a, b)th component of $\mathbf{W}_U$ and $\mathbf{W}_L$ are the weighting factors $w_{Ua}(\omega_b)$ and $w_{La}(\omega_b)$, respectively, as defined in (1). $w_{Ua}(\omega_b)$ or $w_{La}(\omega_b)$ is zero if no upper or lower specification is imposed on $F_a(\boldsymbol{\phi}, \omega_b)$. The coding scheme relating the index of f_j to the indices of nonzeros in $\mathbf{W}_U$ and $\mathbf{W}_L$ are constructed by systematically scanning through $\mathbf{W}_U$ and then $\mathbf{W}_L$, respectively.

IV. AN AUTOMATIC DECOMPOSITION ALGORITHM

An automatic decomposition algorithm for optimization of microwave systems has been developed and implemented. The algorithm heuristically decides when to update the sensitivity matrix and the decomposition dictionary. The formulation and the sequence of suboptimization problems are dynamically determined. The degree of decomposition is reduced as the system converges to its overall solution. As a special case, if all variables interact with all functions, our approach solves only one subproblem, this being identical to the original overall optimization.

Step 1 Initialize sparse factor λ. Calculate the sensitivity matrix **S** and the decomposition dictionary **D**. Calculate **f**.

Comment The initial sensitivity matrix can be obtained from a suitable Monte-Carlo sensitivity analysis performed off-line. All error functions are calculated in this step.

Step 2 Define ℓ such that

$$f_{\text{worst}} = \max_{j \in J_\ell} f_j = \max_{j \in J} f_j .$$

Comment The ℓth function group contains the worst response. Such a function group will be frequently chosen as the reference group to be penalized.

Step 3 For the given ℓ, determine the sequence of candidate rows in **D**. Rank the candidates in decreasing priority. Set $k = 0$.

Comment The ℓth function group is the reference group to be penalized. All variable groups correlating with the ℓth function group are considered as candidates.

Step 4 If $k = 0$ then set k to the row index of the first candidate, otherwise set k to the row index of the next candidate. If such a candidate does not exist then go to Step 8.

Comment The candidate groups of variables are sequentially selected. Each entry into this step results in a selection of a candidate with a lower priority than the current one.

Step 5 Define I^s and J^s using the current k, ℓ. If I^s and J^s are identical with their previous values then go to Step 4. Solve the suboptimization problem of (5).

Terminate the optimization if'

$$\max_{j \notin J^s} \; f_j \; > \; \lambda' \; f_{worst} \; .$$

Comment A subproblem is formulated and solved in this step. By checking the functions not covered in the present suboptimization, any significant deterioration in the overall objective function is prevented. The factor λ' can be, e.g., 1.2.

Step 6 If $I^s = I$ and $J^s = J$ then stop.

Comment The program terminates following the completion of an overall optimization which is considered as the last subproblem.

Step 7 Calculate **f**. Calculate

$$f_{worst} \; = \; \max_{j \in J} \; f_j \; .$$

Go to Step 5.

Comment An overall simulation is performed. By going to Step 5, the current reference function group can be continuously penalized in the next subproblem even if this group does not include the worst error functions.

Step 8 If

$$\max_{j \in J^s} \; f_j \; < \; \max_{j \in J} \; f_j$$

then go to Step 2. If $\lambda \approx 0$ then stop otherwise, update **S**, reduce λ, update dictionary **D** and go to Step 3.

Comment: When the selection of a candidate fails, a new sequence of candidates will be defined by going to Steps 2 or 3. By reducing the sparse factor λ, the degree of decomposition is reduced as the overall solution is being approached. The reference function group will be readjusted if the existing one does not contain the maximum error function. For completely decomposable problems, the terminating conditions in Step 6 will not be satisfied and the program will exit from Step 8.

While the theory in the previous sections is applicable to general optimization problems such as the least pth optimization, the algorithm described in this section is particularly suitable for the minimax optimization defined by (4). Both variables and functions are allowed to overlap between different subproblems. Convergence of the algorithm is not theoretically guaranteed.

V. LARGE SCALE OPTIMIZATION OF MULTIPLEXERS

The automatic decomposition technique was tested on the optimization of microwave multiplexers used in satellite communications. Specifications were imposed on the common port return loss and individual channel insertion loss functions. Each suboptimization was solved using a recent minimax algorithm[22]. Until our recent paper on multiplexers[2], the reported design and manufacturing of these devices were limited to 12 channels[23-27].

A contiguous band 5-channel multiplexer was specifically optimized to illustrate the novel process of automatic decomposition, as shown in Fig. 3. Functions associated with the same channel are grouped together. Variables for each channel include 6 coupling parameters, 6 cavity resonances, input and output transformer ratios (n_1 and n_2) and the distance measure from the channel filter to the short circuit main cascade termination. The overall problem involved 75 variables and 124 nonlinear functions. Figs. 3(a)-(d) show the multiplexer responses for the first 3 suboptimizations. Eleven suboptimizations were used reaching the optimal solution shown in Fig. 3(e). The final subproblem was the overall optimization.

We also tested our approach on a 16-channel multiplexer involving 240 variables and 399 nonlinear functions. The responses at the starting point is shown in Fig. 4. Only 10 suboptimizations were performed before reaching the response of Fig. 5. Then a full optimization was activated resulting in all responses satisfying their specifications as shown in Fig. 6. A comparison between the optimal design with and without decomposition is provided in Table III. When used to obtain a good starting point for subsequent optimization, the decomposition approach offers considerable reductions in both CPU time and storage. The feasibility of obtaining a near optimum for large problems using computers with memory limitations is observed from the table. Such a near optimum is obtained at the cost of increased CPU time. When close to the desired solution, the sizes of the subproblems may approach that of the overall problem. In this case, the performance of optimization does not differ significantly with or without decomposition, unless the original problem is almost completely decomposable.

VI. CONCLUSION

We have presented an automated decomposition approach for optimization of large microwave systems. Compared with the existing decomposition methods, the novelty of our approach lies in its generality in terms of device independency and its automation. Advantages of the approach are 1) a very significant saving of CPU time and/or computer storage and 2) efficient decomposition by automation. By partitioning the overall problem into smaller ones, the approach promises to provide a basis for computer-assisted tuning. It contributes positively towards future general computer software for large-scale optimization of microwave systems.

ACKNOWLEDGEMENTS

The authors acknowledge fruitful interactions with Dr. W. Kellermann, now with Ontario Hydro, Toronto, Canada, and Dr. S. Daijavad, now with the University of California, Berkeley, CA. Mr. S.H. Chen and Ms. M. L. Renault of McMaster University, Hamilton, Canada, provided stimulating suggestions and cooperation throughout this project. Technical discussions with Drs. M.H. Chen of TRW, Redondo Beach, CA, and K. Madsen of the Technical University of Denmark, Lyngby, Denmark, are also appreciated.

REFERENCES

[1] H. Kondoh, "An accurate FET modelling from measured S-parameters", IEEE Int. Microwave Symp. Digest (Baltimore, MD), pp. 377-380, 1986.

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

[3] D.M. Himmelblau, Ed., Decomposition of Large-Scale Problems. Amsterdam: North-Holland, 1973.

[4] A.M. Geoffrion, "Elements of large-scale mathematical programming", Management Science, vol. 16, pp. 652-691, 1970.

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

[6] H.P.L. Luna, "A survey on informational decentralization and mathematical programming decomposition", in Mathematical Programming, R.W. Cottle, M.L. Kelmanson and B. Korte Eds. Amsterdam: Elsevier, 1984.

[7] F. Luccio and M. Sami, "On the decomposition of networks in minimally interconnected subnetworks", IEEE Trans. Circuit Theory, vol. CT-16, pp. 184-188, 1969.

[8] L.O. Chua and L.K. Chen, "Diakoptic and generalized hybrid analysis", IEEE Trans. Circuits Syst., vol. CAS-23, pp. 694-705, 1976.

[9] F.F. Wu, "Solution of large-scale networks by tearing", IEEE Trans. Circuits Syst., vol. CAS-23, pp. 706-713, 1976.

[10] M.D. Tong and W.K. Chen, " Hybrid analysis of a large-scale network by node-tearing", Proc. IEEE Int. Symp. Circuits and Systems (San Jose, CA), pp. 178-181, 1986.

[11] H. Asai, M. Urano and M. Tanaka, "A new simulation technique for a large scale transistor circuit based on relaxation method using network tearing", Proc. IEEE Int. Symp. Circuits and Systems (San Jose, CA), pp. 161-164, 1986.

[12] J. Pospisil, "Linear network analysis using scattering matrix decomposition", Proc. IEEE Int. Symp. Circuits and Systems (San Jose, CA), pp. 193-196, 1986.

[13] A. Sangiovanni-Vincentelli, L.K. Chen and L.O. Chua, "An efficient heuristic cluster algorithm for tearing large-scale networks", IEEE Trans. Circuits Syst., vol. CAS-24, pp. 709-717, 1977.

[14] A.E. Salama, J.A. Starzyk and J.W. Bandler, "A unified decomposition approach for fault location in large analog circuits", IEEE Trans. Circuits Syst., vol. CAS-31, pp. 609-622, 1984.

[15] S.N. Talukdar, T.C. Giras and V.K. Kalyan, "Decompositions for optimal power flows", IEEE Trans. Power Apparatus Syst., vol. PAS-102, pp. 3877-3884, 1983.

[16] G.C. Contaxis, C. Delkis and G. Korres, "Decoupled optimal load flow using linear or quadratic programming", IEEE Trans. Power Syst., vol. PWRS-1, pp. 1-7, 1986.

[17] K.L. Lo and Y.M. Mahmoud, "A decoupled linear programming technique for power system state estimation", IEEE Trans. Power Syst., vol. PWRS-1, pp. 154-160, 1986.

[18] R. Billinton and S.S. Sachdeva, "Real and reactive power optimization by suboptimum techniques", IEEE Trans. Power Apparatus Syst., vol. PAS-92, pp. 950-956, 1973.

[19] C. Tsironis and R. Meierer, "Microwave wide-band model of GaAs dual gate MESFET's", IEEE Trans. Microwave Theory Tech., vol. MTT-30, pp. 243-251, 1982.

[20] P.J. Courtois, Decomposability. New York, NY: Academic Press, 1977.

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

[22] 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, pp. 1519-1530, 1985.

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

[24] R.G. Egri, A.E. Williams and A.E. Atia, "A contiguous-band multiplexer design", IEEE Int. Microwave Symp. Digest (Boston, MA), pp.86-88, 1983.

[25] R. Tong and D. Smith, "A 12-channel contiguous-band multiplexer for satellite application", IEEE Int. Microwave Symp. Digest (San Francisco, CA), pp. 297-298, 1984.

[26] S.C. Holme, "A 12 GHz 12 channel multiplexer for satellite applications", IEEE Int. Microwave Symp. Digest (San Francisco, CA), pp. 295-296, 1984.

[27] 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, pp. 93-102, 1986.

TABLE I

THE C MATRIX FOR THE FET MODEL

		Function Groups			
Frequency Band	Variables	S_{11}	S_{21}	S_{12}	S_{22}
	g_m	18.55	100.00	87.55	68.33
	C_{gs}	100.00	89.74	67.98	62.25
entire	C_{ds}	4.88	67.74	45.73	100.00
band	C_{dg}	4.24	48.88	100.00	81.27
	R_s	35.53	37.14	100.00	5.88
	R_{ds}	17.44	97.68	70.51	100.00
	τ	31.91	100.00	36.61	59.31
	R_g	100.00	50.67	24.87	29.89
upper	R_d	34.65	74.31	85.85	100.00
half	R_i	100.00	65.63	88.43	39.53
band	L_g	100.00	87.85	57.16	37.44
	L_d	9.99	97.88	61.78	100.00
	L_s	62.94	31.31	100.00	21.99

TABLE II

NORMALIZED DECOMPOSITION DICTIONARY **D**

		Function Groups			
Frequency Band	Variable Groups	S_{11}	S_{21}	S_{12}	S_{22}
entire band	R_{ds}, C_{ds}	0.00	0.00	0.00	1.00
	C_{gs}	1.00	0.00	0.00	0.00
	C_{dg}, R_s	0.00	0.00	1.00	0.00
	g_m	0.00	1.00	0.00	0.00
upper half band	R_d, L_d	0.00	0.00	0.00	1.00
	R_g, R_i, L_g	1.00	0.00	0.00	0.00
	L_s	0.00	0.00	1.00	0.00
	τ	0.00	1.00	0.00	0.00

TABLE III

COMPARISON OF 16-CHANNEL MULTIPLEXER OPTIMIZATION WITH AND WITHOUT DECOMPOSITION

Purpose of Optimization+	Reduction in Objective Function	Criteria for Comparison	With Decomp.	Without Decomp.
to provide a good starting point for further optimization	from 13.46 to 2.4	CPU time *	99	250
		working space needed†	2,197	483,036
to obtain a near optimum solution	from 13.46 to 0.32	CPU time *	651	553
		working space needed†	73,972	483,036
to obtain optimum solution	from 13.46 to −0.09	CPU time *	1045	1289
		working space needed†	483,036	483,036

+ different sparse factors λ have been used to control the degree of decomposition for the three different purposes.

* seconds on the FPS-264 mainframe.

† of machine memory units (one unit per real number) required by the minimax optimization package[22].

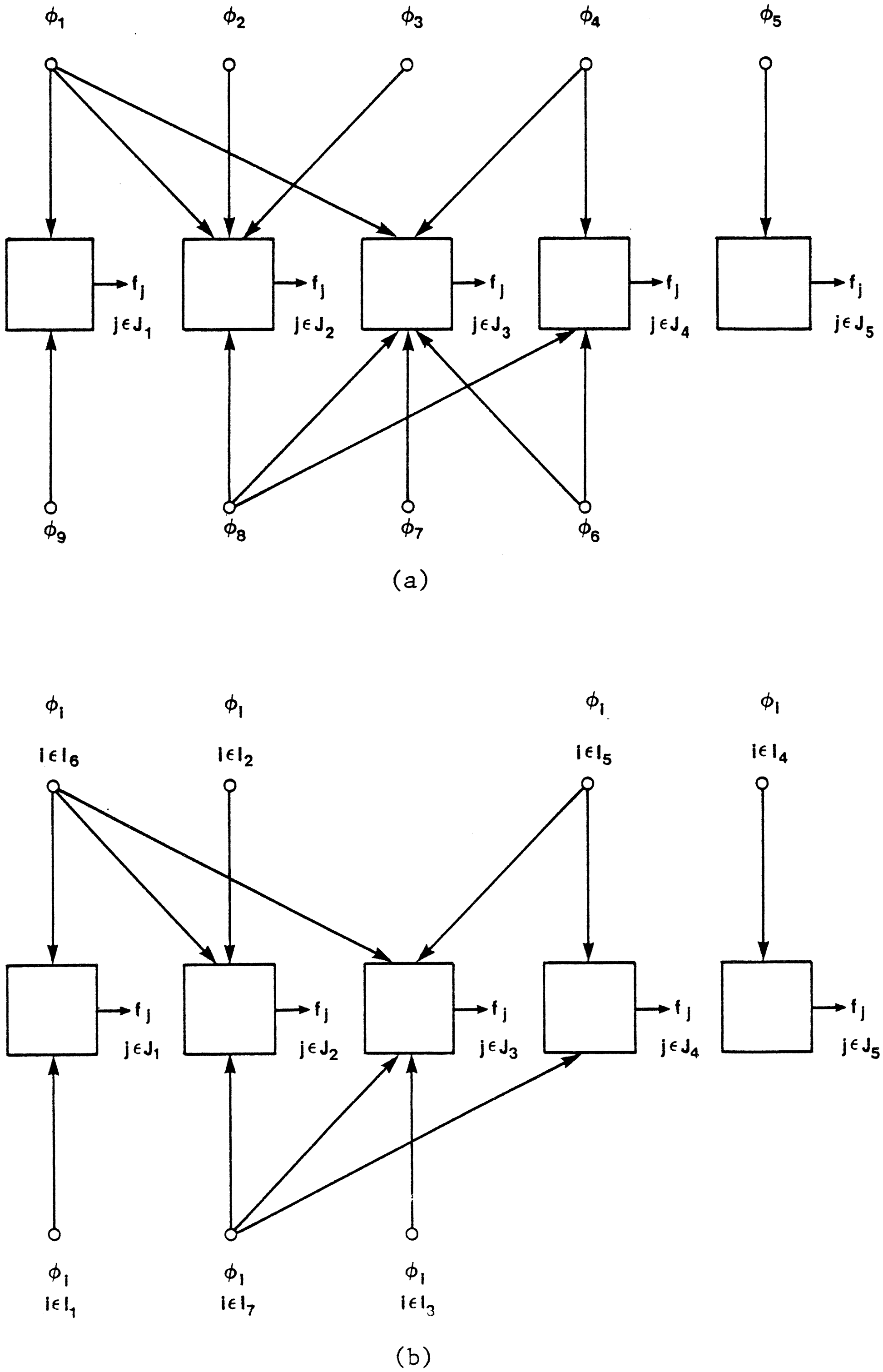


Fig. 1 A fictitious example showing only the strong interconnections between variables and function groups.
(a) System configuration corresponding to matrix **C**.
(b) System configuration corresponding to matrix **D**.

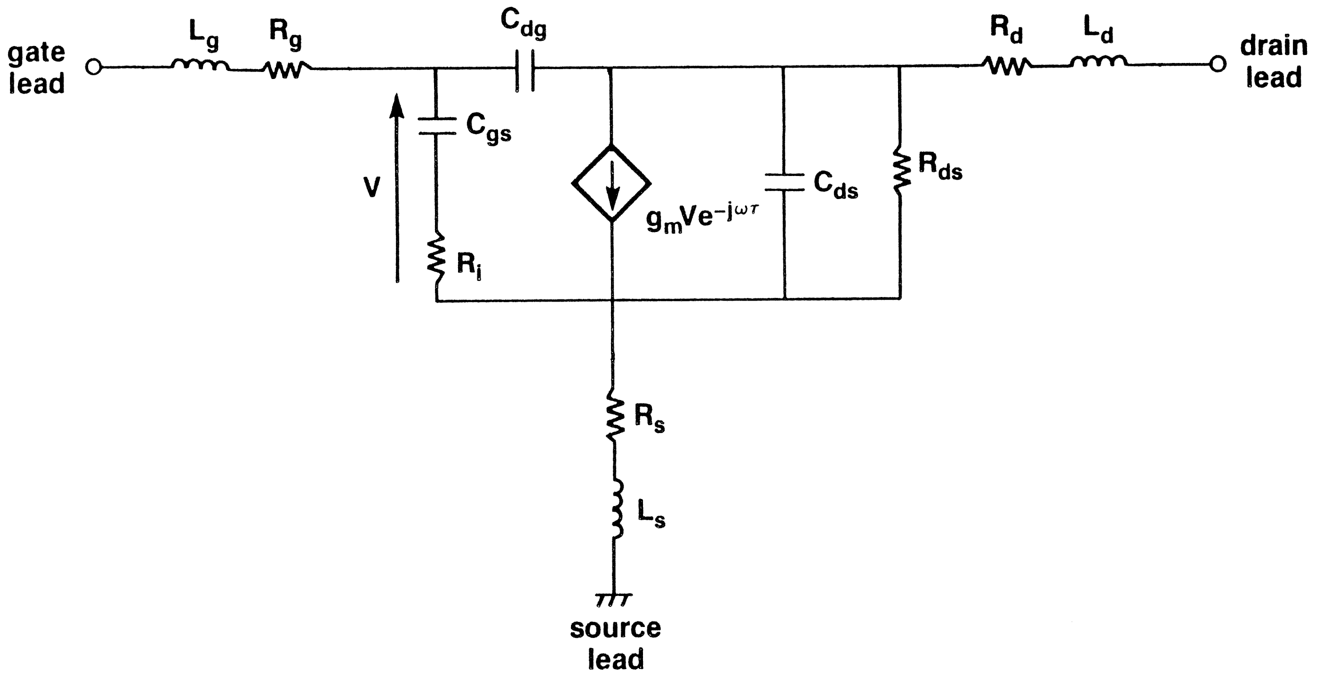
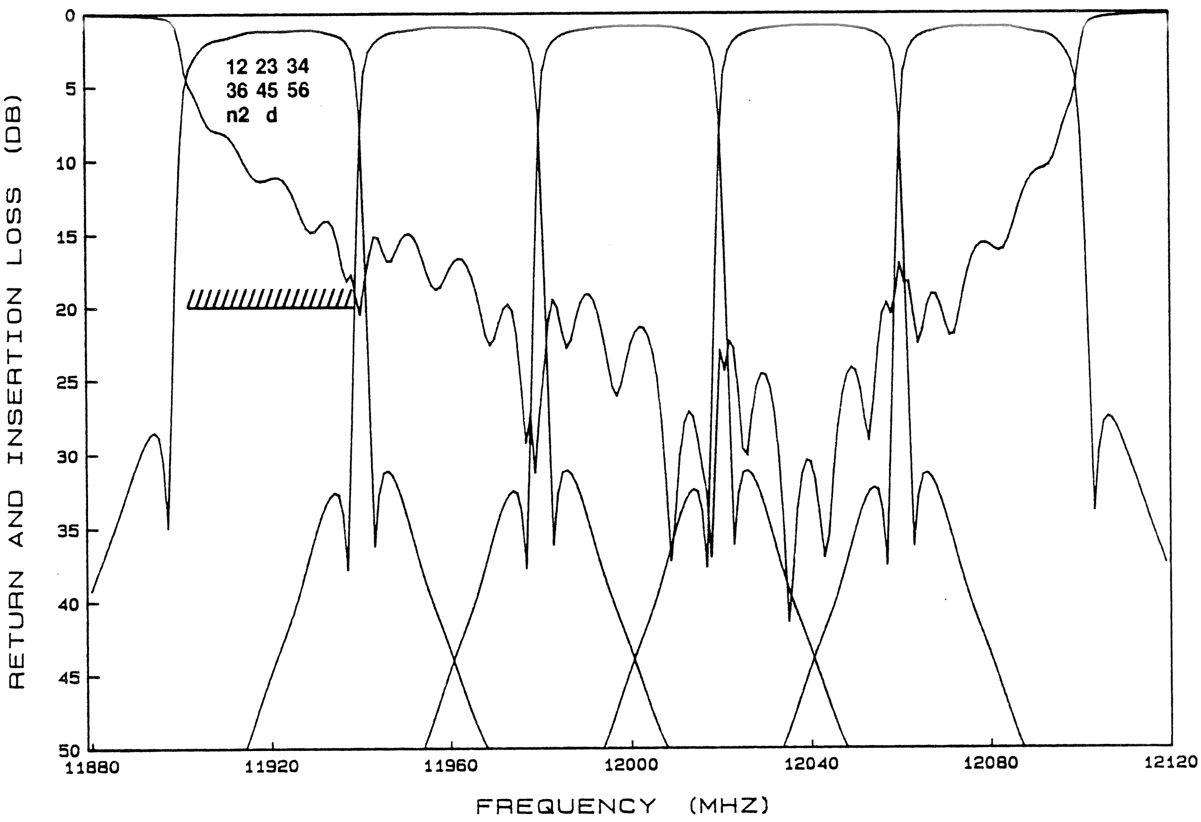


Fig. 2 A FET equivalent circuit.

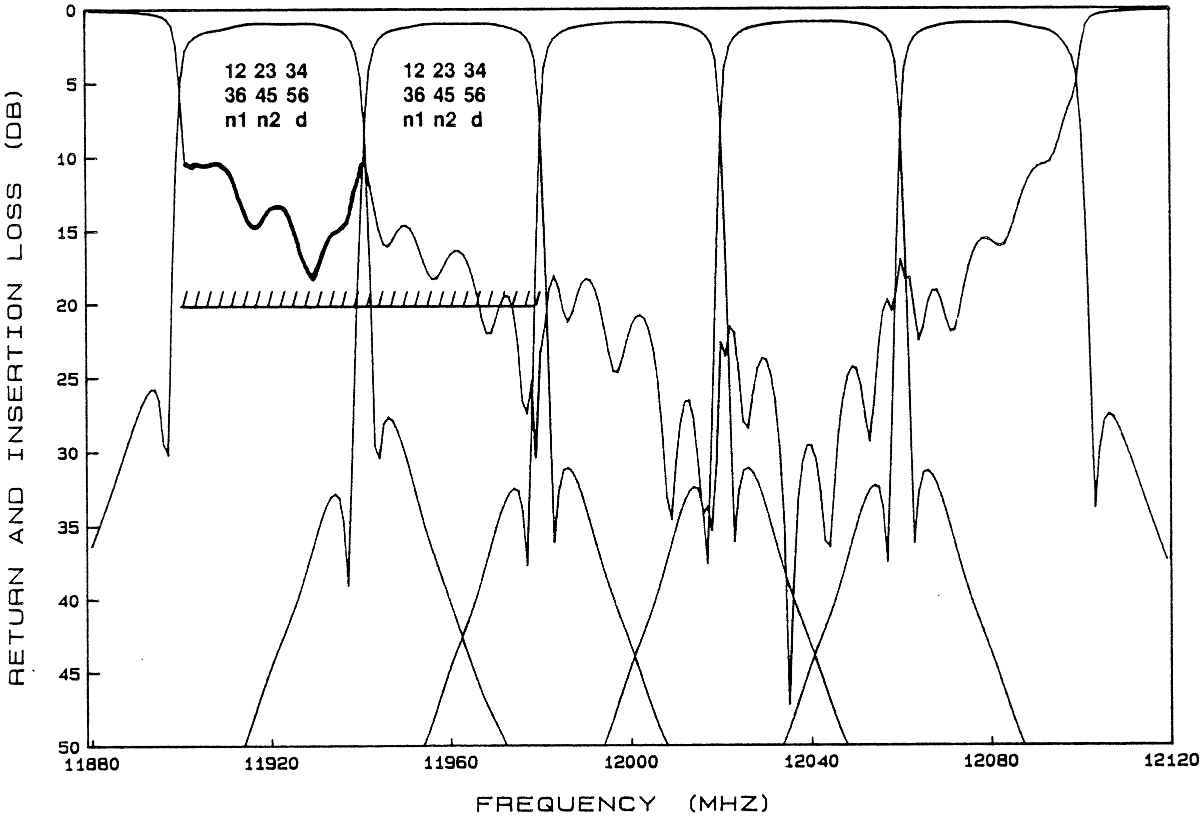
Fig. 3 Return and insertion loss responses of the 5-channel multiplexer for each suboptimization. The 20 dB specification line indicates which channel(s) is to be optimized in the next subproblem. The variables to be selected are indicated, e.g., 35 representing coupling M_{35}, d representing the distance of the corresponding channel filter from the short circuit main cascade termination. The previously optimized channels are highlighted by thick response curves.

(a) Responses at the starting point.
(b) Responses after the 1st suboptimization.
(c) Responses after the 2nd suboptimization.
(d) Responses after the 3rd suboptimization.
(e) Responses at the optimum solution.



RETURN AND INSERTION LOSS (DB)
0
5
10
15
20
25
30
35
40
45
50
11880
11920
11960
12000
12040
12080
12120
FREQUENCY (MHZ)
12 23 34
36 45 56
n2 d

Fig. 3(a)



RETURN AND INSERTION LOSS (DB)
0
5
10
15
20
25
30
35
40
45
50
11880
11920
11960
12000
12040
12080
12120
FREQUENCY (MHZ)
12 23 34
36 45 56
n1 n2 d
12 23 34
36 45 56
n1 n2 d

Fig. 3(b)

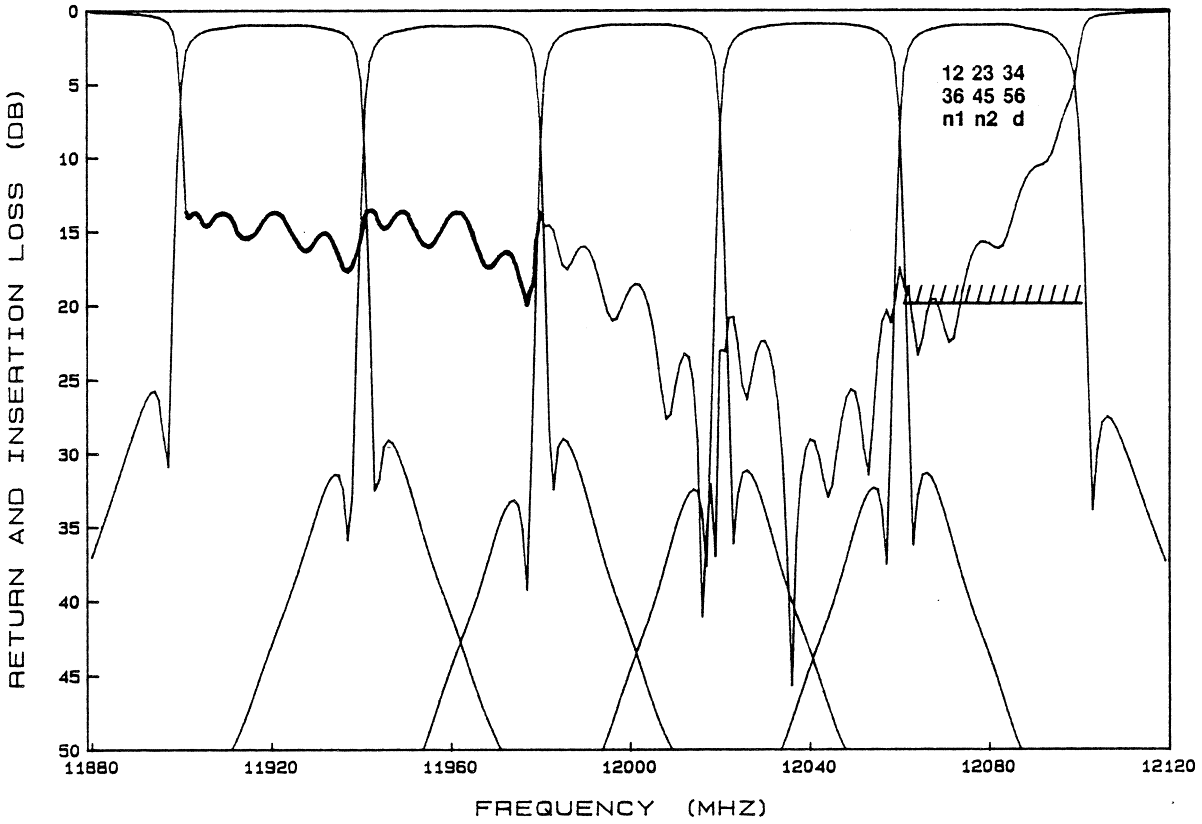

RETURN AND INSERTION LOSS (DB)
0
5
10
15
20
25
30
35
40
45
50
11880
11920
11960
12000
12040
12080
12120
FREQUENCY (MHZ)
12 23 34
36 45 56
n1 n2 d

Fig. 3(c)

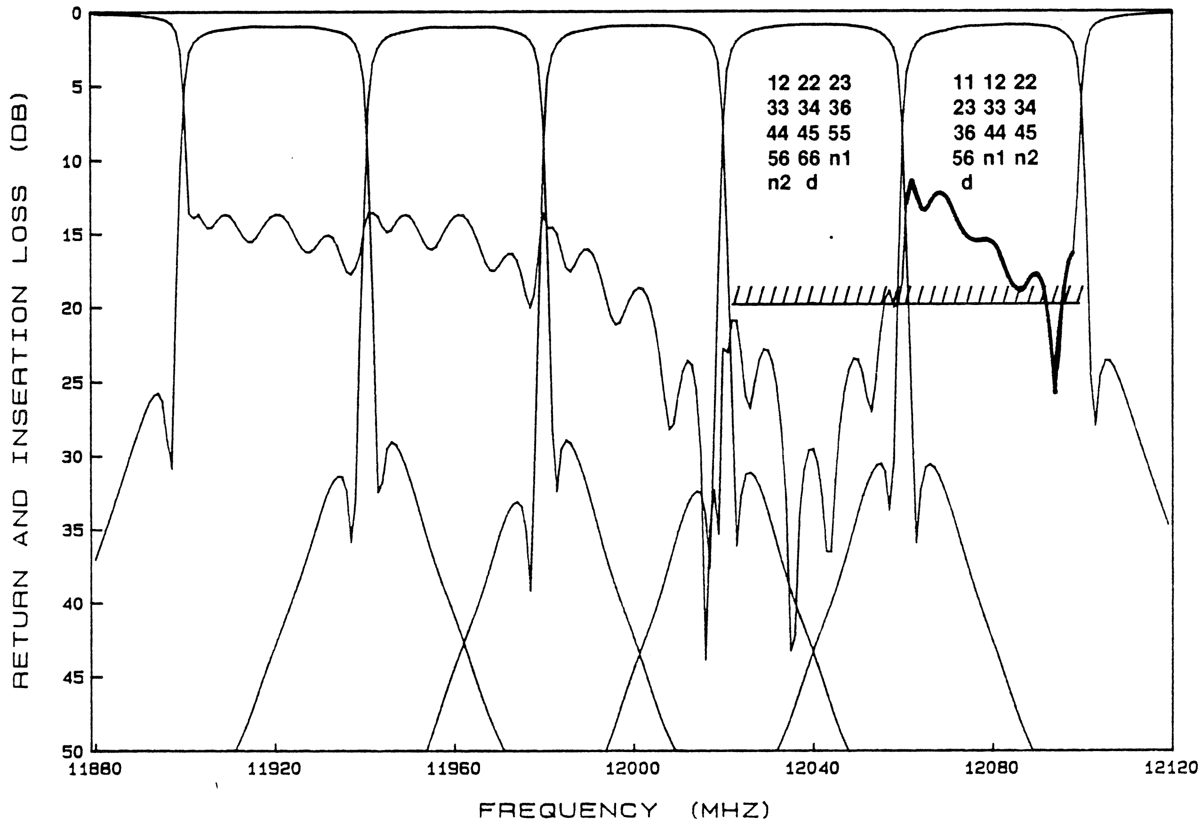

RETURN AND INSERTION LOSS (DB)
0
5
10
15
20
25
30
35
40
45
50
11880
11920
11960
12000
12040
12080
12120
FREQUENCY (MHZ)
12 22 23
33 34 36
44 45 55
56 66 n1
n2 d
11 12 22
23 33 34
36 44 45
56 n1 n2
d

Fig. 3(d)

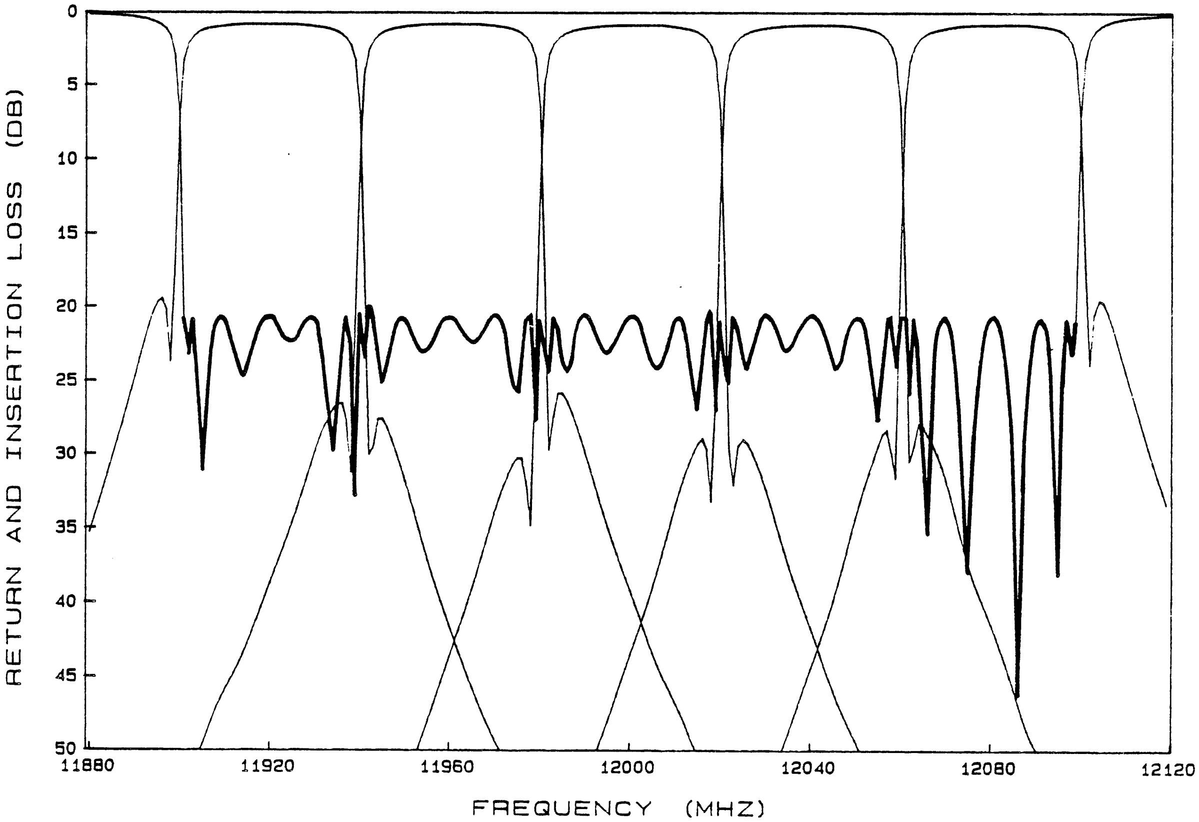

RETURN AND INSERTION LOSS (DB)
0
5
10
15
20
25
30
35
40
45
50
11880
11920
11960
12000
12040
12080
12120
FREQUENCY (MHZ)

Fig. 3(e)

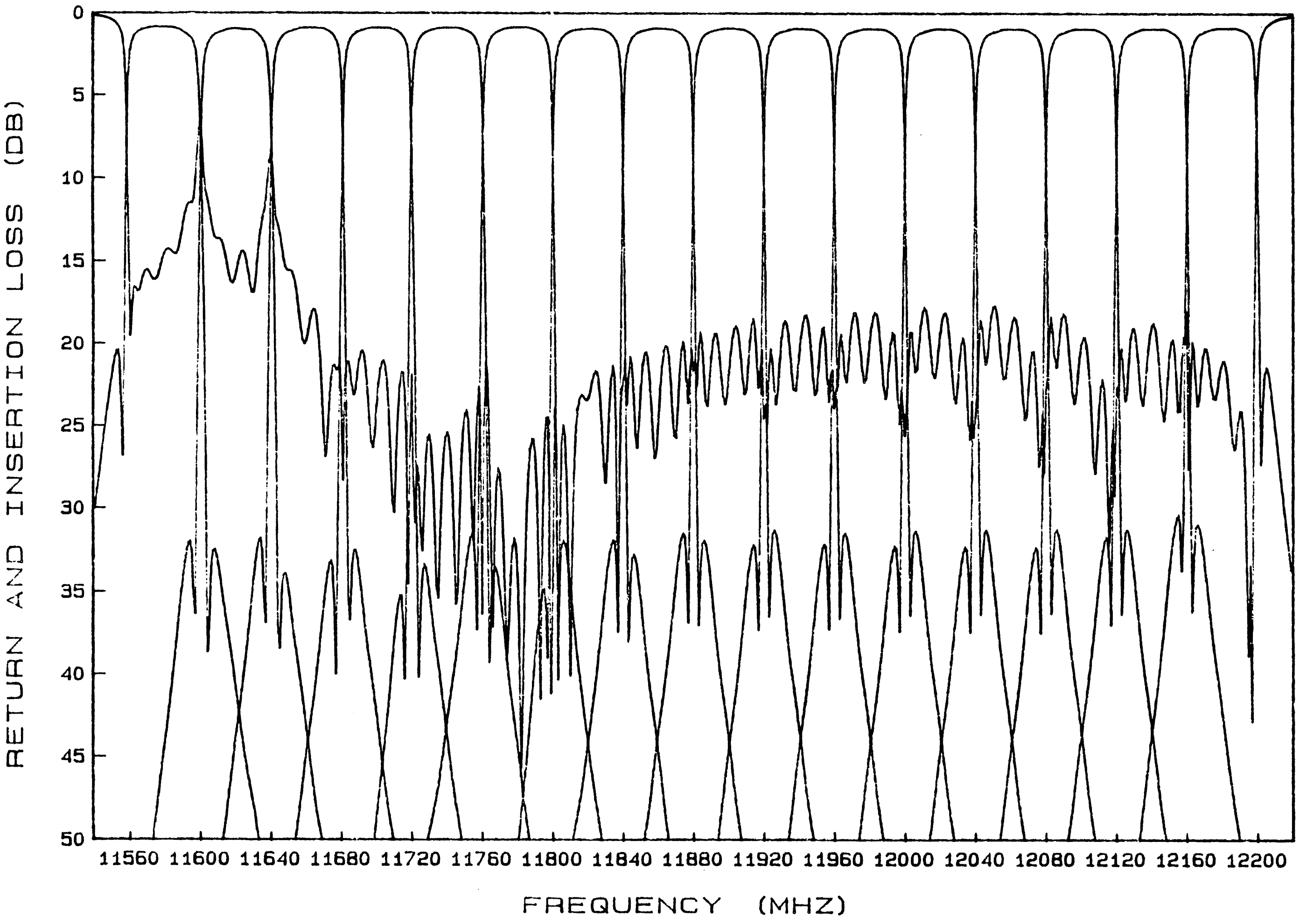


Fig. 4 Return and insertion loss responses of the 16-channel multiplexer before optimization.

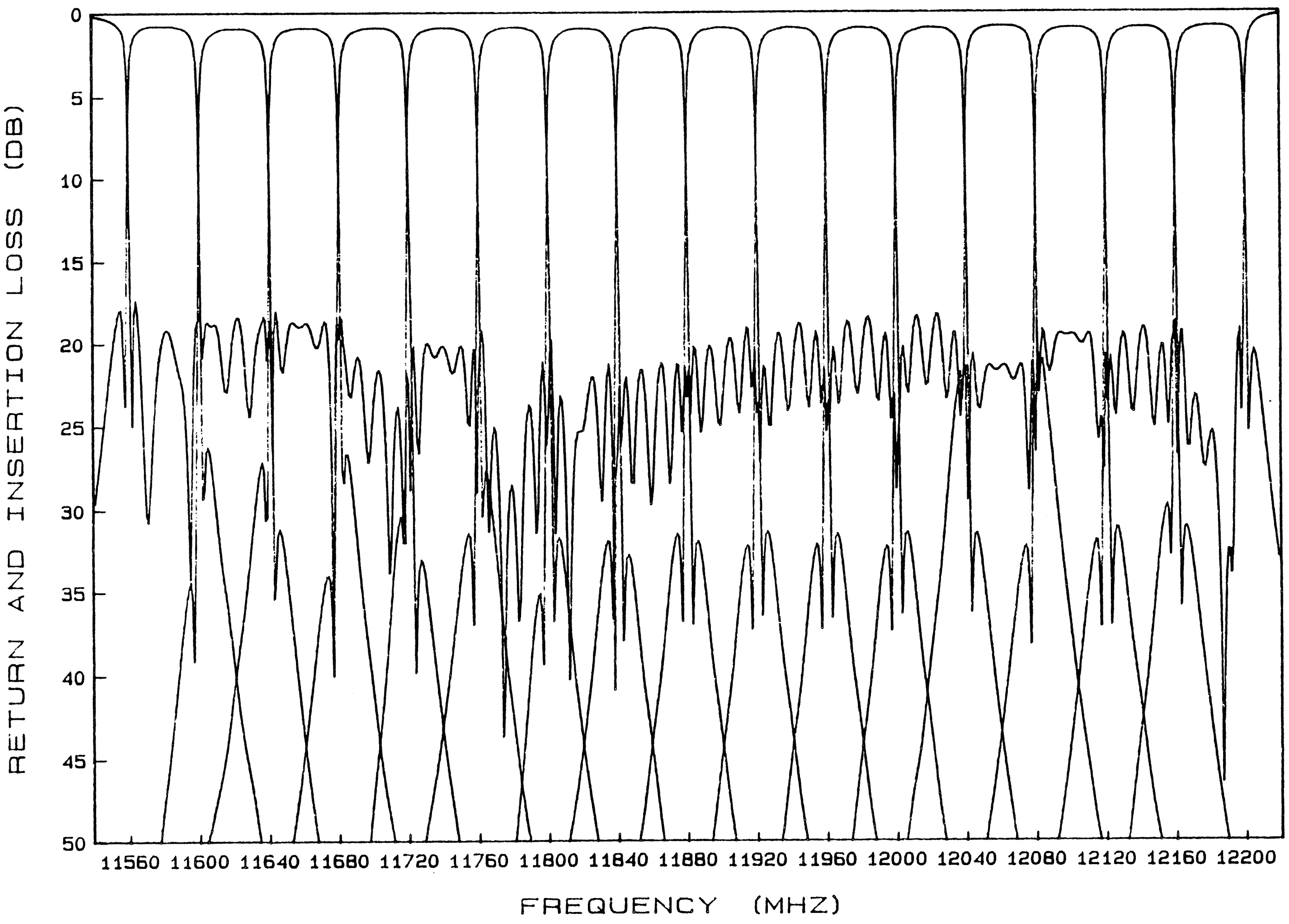


Fig. 5 Return and insertion loss responses of the 16-channel multiplexer after 10 suboptimizations. Each of the 10 suboptimizations involved responses associated with only one channel and no more than 15 variables.

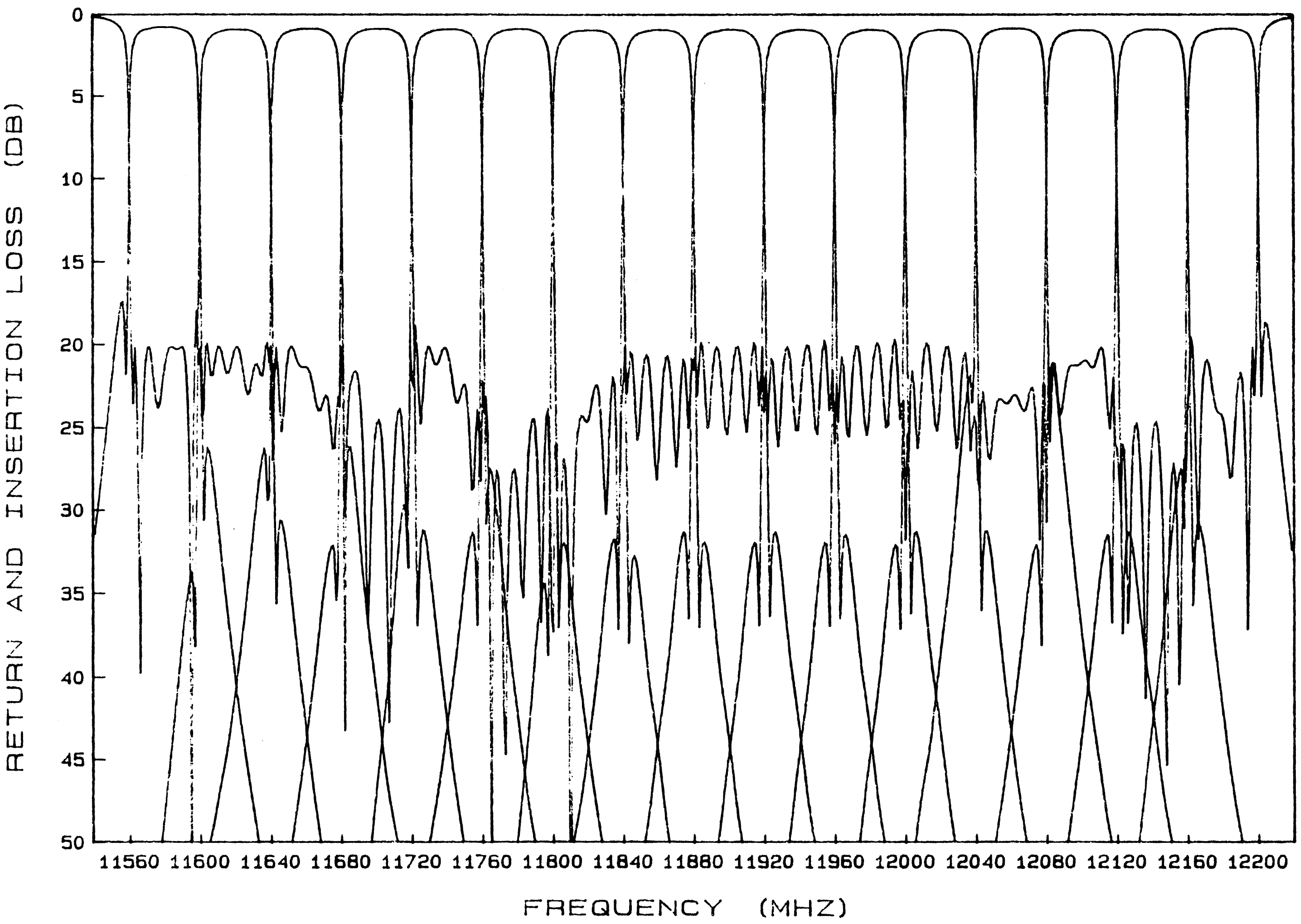


Fig. 6 Return and insertion loss responses of the 16-channel multiplexer at the overall solution. All design specifications are satisfied.

EFFICIENT OPTIMIZATION WITH INTEGRATED GRADIENT APPROXIMATIONS

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

S. Daijavad, Member, IEEE, and K. Madsen

Abstract A flexible and effective algorithm is proposed for efficient optimization with integrated gradient approximations. It combines the techniques of perturbations, the Broyden update and the special iterations of Powell. Perturbations are used to provide an initial approximation as well as regular corrections. The approximate gradient is updated using Broyden's formula in conjunction with the special iterations of Powell. A modification to the Broyden update is introduced to exploit possible sparsity of the Jacobian. Utilizing this algorithm, powerful gradient-based nonlinear optimization tools for circuit CAD can be employed without the effort of calculating exact derivatives. Applications of practical significance are demonstrated. The examples include robust small signal FET modeling using the ℓ_1 techniques and simultaneous processing of multiple circuits, worst-case design of a microwave amplifier as well as minimax optimization of a 5-channel manifold multiplexer. Computational efficiency is greatly improved as compared to estimating derivatives entirely by pert''rbations.

This work was supported in part by the Natural Science and Engineering Research Council of Canada under Grants A7239 and G1135, and in part by Optimization Systems Associates Inc.

J.W. Bandler and S.H. Chen 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.

S. Daijavad is with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, California 94720, U.S.A.

K. Madsen is with the Institute for Numerical Analysis, Technical University of Denmark, Building 302, DK-2800 Lyngby, Denmark.

I. INTRODUCTION

Many powerful gradient-based algorithms have been developed in recent years for nonlinear optimization and applied to circuit CAD problems. For example, Bandler, Kellermann and Madsen have described algorithms for linearly constrained minimax and ℓ_1 optimization [1], [2]. However, the effort to extend their application to a wide range of practical problems has been frustrated by the requirement of exact gradients of all functions with respect to all variables. For some applications, either an explicit sensitivity expression is not available, e.g., when time-domain analysis and nonlinear circuits are involved, or the actual evaluation of such an expression is very tedious and time-consuming, e.g., for large-scale networks. Partly due to these difficulties, exact sensitivity calculations have not been implemented in many 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. The inability or inconvenience in calculating the exact derivatives has created a gap between the theoretical advances in gradient-based nonlinear optimization techniques and their actual implementation.

With only the function values available, as is the case for many CAD packages on the market, one usually resorts to the method of perturbations (finite differences) for gradients. However, this seemingly simple alternative becomes extremely inefficient when large-scale problems have to be dealt with.

In this paper, we propose a flexible and effective approach to optimization with integrated gradient approximations. It is a hybrid approach which incorporates the use of perturbations, the Broyden update [3] and the special iterations of Powell [4]. The proposed algorithm extends the previous work by Madsen [5] and Zuberek [6] in two aspects. Perturbations are integrated in a flexible manner to allow regular corrections to the approximate gradients. Therefore, a suitable compromise between accuracy and computational labor may be achieved for various applications, especially for large-scale circuit optimization. We also propose a modified Broyden update to take advantage of a possible sparse structure of the problem.

The practical usefulness of the new algorithm is demonstrated through three diverse applications. The subjects are of primary interest to microwave circuit engineers: robust small signal modeling of FET devices, worst-case fixed tolerance design of a microwave amplifier and large-scale optimization of manifold multiplexers. Applying an approach to robust device modeling proposed by the authors [7] which employs the ℓ_1 optimization techniques and a novel concept of simultaneous processing of multiple circuits, we have obtained self-consistent models of a FET device using real measurement data. By integrating gradient approximations with a powerful minimax algorithm [1], we are able to optimize a 5-channel noncontiguous band multiplexer efficiently and without exact derivatives. The multiplexer problem involves 75 nonlinear variables.

II. GRADIENT APPROXIMATIONS

A. Estimating the Gradient by Perturbations

The first-order derivative of $f_j(x)$ with respect to x_i can be estimated by

$$\frac{\partial f_j(x)}{\partial x_i} \approx \frac{f_j(x+h\mathbf{u}_i) - f_j(x)}{h} , \tag{1}$$

where $x = [x_1 \ x_2 \ \dots \ x_n]^T$ is the vector of variables, and $\mathbf{u}_i$ is a column vector which has 1 in the ith position and zeros elsewhere. The accuracy of such an estimate may be improved by using a smaller h as well as by averaging the results of a two-sided approximation (using both positive and negative perturbations). This method is straight forward and reliable. However, the computational labor involved grows in proportion to the dimension of the problem.

In the new algorithm described in this chapter, perturbations are used to obtain an initial approximation to the gradient at the starting point of an optimization process. During the optimization, we may also incorporate a regular use of perturbations to maintain the accuracy of gradient approximations at a desirable level.

B. The Broyden Update

The Broyden update refers to a rank-one formula proposed by Broyden [3] as

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \frac{\mathbf{f}(\mathbf{x}_k+\mathbf{h}_k) - \mathbf{f}(\mathbf{x}_k) - \mathbf{G}_k\mathbf{h}_k}{\mathbf{h}_k^T\mathbf{h}_k}\,\mathbf{h}_k^T\,, \tag{2}$$

where $\mathbf{G}_k$ is an approximation of the Jacobian $[\partial \mathbf{f}^T/\partial \mathbf{x}]^T$ at $\mathbf{x}_k$, $\mathbf{h}_k$ is an increment vector and $\mathbf{G}_{k+1}$ provides an updated Jacobian. The values of the function f at $\mathbf{x}_k$ and $(\mathbf{x}_k+\mathbf{h}_k)$ are assumed available. If the two points ($\mathbf{x}_k$ and $(\mathbf{x}_k+\mathbf{h}_k)$) are iterates of the optimization process, then the Broyden update requires no additional function evaluations, regardless of the dimension of the problem.

Apparently, the approximate Jacobians generated by the Broyden update are in general less accurate as compared with those obtained from perturbations. Hence, the optimization may require more steps to reach the solution or may not reach the correct solution at all. Broyden [3] has shown that for quadratic functions the Broyden update will converge and will reduce the overall computational effort. Although such properties can not be proved for a general nonlinear problem, the Broyden update still provides an efficient alternative for approximating derivatives.

The updated approximation $\mathbf{G}_{k+1}$ satisfies the following equation

$$\mathbf{f}(\mathbf{x}_k+\mathbf{h}_k) - \mathbf{f}(\mathbf{x}_k) = \mathbf{G}_{k+1}\mathbf{h}_k\ . \tag{3}$$

In other words, $\mathbf{G}_{k+1}$ provides a perfect linear interpolation between the two points $\mathbf{x}_k$ and $(\mathbf{x}_k+\mathbf{h}_k)$.

Some difficulties in the application of the Broyden update have been observed by many researchers (see, for example, [4], [5] and [6]).

(1) If some functions are linear in some variables and if the corresponding components of $\mathbf{h}_k$ are nonzero, then the approximation of constant derivatives are updated by nonzero values. Consider a simple example. Let $f_j = x_1^2 + 2x_3$ be a function in $\mathbf{f}$. Denote the variables by $\mathbf{x} = [x_1\ x_2\ x_3]^T$ and the gradient by $\mathbf{f}_j'(\mathbf{x}) = [g_1\ g_2\ g_3]^T$. Two components of the gradient, namely $g_2 = 0$ and $g_3 = 2$, are constants and can be found accurately by perturbations. g_1 is the only component that needs to be updated. Suppose that $\mathbf{x}_k = [1\ \ 1\ \ 1]^T$, $\mathbf{h}_k = [0.5\ \ 0.5\ \ 0.5]^T$ and a perfect estimation of $\mathbf{f}_j'(\mathbf{x}_k)$ is available as $[2\ \ 0\ \ 2]^T$. The approximation to $\mathbf{f}_j'(\mathbf{x}_k+\mathbf{h}_k)$, as given by the Broyden update, would be $[2.167\ \ 0.167\ \ 2.167]^T$ (the true value is $[3\ \ 0\ \ 2]^T$).

(2) Along directions orthogonal to $\mathbf{h}_k$ the Jacobian is not updated:

$$\mathbf{G}_{k+1}\mathbf{p} = \mathbf{G}_k\mathbf{p}, \quad \text{for } \mathbf{p}^T\mathbf{h}_k = 0. \tag{4}$$

To overcome these difficulties, we implement a weighted update and the special iterations of Powell [4].

C. Weighted Broyden Update

The weighted update is to be applied to the Jacobian matrix on a row-by-row basis. The jth row vector of the approximate Jacobian, denoted by $(\mathbf{g}_j)_k$, is an approximation to $\mathbf{f}_j'(\mathbf{x}_k)$, the gradient of f_j. Suppose that the Hessian of f_j is available to us and denoted by $\mathbf{H}_j$, then

$$\mathbf{f}_j'(\mathbf{x}_k+\mathbf{h}_k) \approx \mathbf{f}_j'(\mathbf{x}_k) + \mathbf{H}_j(\mathbf{x}_k)\ \mathbf{h}_k. \tag{5}$$

Analogously to (5), we devise an updating formula to obtain an approximation to $\mathbf{f}_j'(\mathbf{x}_k+\mathbf{h}_k)$ as

$$(\mathbf{g}_j)_{k+1} = (\mathbf{g}_j)_k + \alpha\ \mathbf{H}_j(\mathbf{x}_k)\ \mathbf{h}_k. \tag{6}$$

If we choose the coefficient α as

$$\alpha = \frac{f_j(\mathbf{x}_k+\mathbf{h}_k) - f_j(\mathbf{x}_k) - (\mathbf{g}_j)_k^T\mathbf{h}_k}{\mathbf{h}_k^T\,\mathbf{H}_j(\mathbf{x}_k)\,\mathbf{h}_k}\ , \tag{7}$$

then the linear model as given by (3) will be preserved, namely

$$f_j(\mathbf{x}_k+\mathbf{h}_k) - f_j(\mathbf{x}_k) = (\mathbf{g}_j)_{k+1}^T\mathbf{h}_k. \tag{8}$$

In practice we are very unlikely to have access to the Hessian of any f_j. Even so, two basic facts are obvious: the Hessian of a quadratic function is constant, and if f_j is linear in x_i then the ith row and the ith column of the Hessian contain only zeros. Hence, we propose the use of a constant diagonal matrix

$$\mathbf{W}_j = \text{diag}[w_{j1}\ \dots\ w_{jn}],\quad w_{ji} \geq 0,\ \ i = 1,\dots,\ n, \tag{9}$$

This leads to a weighted Broyden update as follows.

$$(\mathbf{g}_j)_{k+1} = (\mathbf{g}_j)_k + \frac{f_j(\mathbf{x}_k+\mathbf{h}_k) - f_j(\mathbf{x}_k) - (\mathbf{g}_j)_k^T\mathbf{h}_k}{\mathbf{q}_{jk}^T\ \mathbf{h}_k}\ \mathbf{q}_{jk}\ ,$$
$$\mathbf{q}_{jk} = \mathbf{W}_j\ \mathbf{h}_k = [w_{j1}h_{k1}\ \dots\ w_{jn}h_{kn}]^T. \tag{10}$$

The weights w_{ji} provide a measure of the linearity of f_j. If f_j is linear in x_i, we set w_{ji}=0, and if f_j is nearly linear in x_i, we assign a small value to w_{ji}. It should be clear from (10) that only the relative magnitude of the weights is important, not their absolute values.

Consider the simple example we have used in the previous section, namely $f_j = x_1^2 + 2x_3$. Since f_j is independent of x_2 and linear in x_3, we set $w_{j2} = w_{j3} = 0$ and $w_{j1} = 1$. The approximate gradient given by (10) is $[2.5\ 0\ 2]^T$, compared to the result given by the Broyden update as $[2.167\ 0.167\ 2.167]^T$, and the true gradient $[3\ 0\ 2]^T$.

The assignment of weights requires some knowledge of the functional relationship of $f_j(\mathbf{x})$. Such a knowledge may come from experience or may be gained from sensitivity analyses by performing a few perturbations. For instance, for a particular circuit, it may be known that some designable parameters have little influence on the performance function over some frequency or time intervals. Using an adaptive method to find $\mathbf{W}_j$ might be of some theoretical interest. But it was felt to be unnecessary and too complicated to be practical at the present time.

D. Powell's Special Iterations

The Broyden update is a rank-one method. As has been shown in (4), along directions orthogonal to $\mathbf{h}_k$ the approximate Jacobian is not updated. If some consecutive steps of optimization happen to be collinear, the updating procedure may not converge. Powell [4] suggested a method which produces strictly linearly independent directions. For this purpose, special iterations are introduced which intervene between the ordinary iterations of optimization. The increment vector of such a special iteration is not calculated to minimize the error functions, instead it serves the purpose of improving the accuracy of gradient approximations. The algorithm for computing the increment vector for a special iteration, as derived by Powell, is given in the Appendix.

III. A HYBRID APPROXIMATION ALGORITHM

Our hybrid algorithm for gradient approximations consists of an initial approximation, the Broyden update, Powell's special iterations and regular corrections provided by perturbations.

At the starting point of optimization, the initial approximate Jacobian $\mathbf{G}_0$ is usually computed by perturbations. However, $\mathbf{G}_0$ may be already available, for example, it may have been stored from a previous optimization, and can be utilized to avoid unnecessary computations. This option would be useful if similar problems are being solved repetitively (e.g., the same circuit is optimized with respect to different specifications). The accuracy of $\mathbf{G}_0$ is not very critical to the overall approximation. We have observed for some examples that convergence was achieved despite the erroneous estimates of $\mathbf{G}_0$.

There is little hard evidence as to how frequently the special iterations should be used. Numerical experience, ours as well as other authors', has suggested the use of a special iteration between every two ordinary ones (i.e., every third iteration is a special iteration). Also, in our implementation, a special iteration is skipped provided that the changes in the functions agree fairly well with the linear prediction by the approximate gradient. This is considered to be true if

$$\| \mathbf{f}_j(\mathbf{x}_k+\mathbf{h}_k) - \mathbf{f}_j(\mathbf{x}_k) - \mathbf{G}_k\mathbf{h}_k \| < 0.1 \| \mathbf{f}_j(\mathbf{x}_k+\mathbf{h}_k) - \mathbf{f}_j(\mathbf{x}_k) \| . \tag{11}$$

The purpose of this provision is to avoid unnecessary computations.

Whether perturbations should be used during optimization depends on the application. For small or mildly nonlinear problems, the Broyden update may suffice. For large-scale problems, especially in circuit applications where highly nonlinear functions are involved, the correction provided by perturbations is likely to be necessary. We have incorporated in our algorithm the use of perturbations with prescribed regularity, say, at every kth optimization iteration.

The Broyden update with or without weights, depending on whether the necessary knowledge of **f**(x) is available, is employed between perturbations.

Software for gradient-based optimization typically requires a user-defined routine which accepts a set of values for x as input and returns the values of f(x) as well as the first-order derivatives. We have implemented an interface which integrates gradient approximations with optimization. Taking a set of values for x from an optimizer, it calls a user-defined routine for the function values, carries out necessary operations for gradient approximations, and then returns to the optimizer the values of **f**(x) as well as the approximate Jacobian.

The interface is transparent to both the optimizer and the user-defined simulation routine. The optimizer is provided with the required gradients, and the user-defined routine (typically a circuit simulation module) works as if the optimizer did not require gradients.

We have integrated our gradient approximation algorithm with two recent optimization methods [1], [2], for the minimax problems as

$$\underset{x}{\text{minimize}} \; \max_{j} \{f_j(x)\} \, , \tag{12}$$

and the ℓ_1 problems as

$$\underset{x}{\text{minimize}} \; \sum_{j=1}^{m} |f_j(x)| \, , \tag{13}$$

respectively. The methods described in [1] and [2] are 2-stage algorithms. The second stage is to be employed near the solution to accelerate the rate of convergence, for which the accuracy. of the approximate gradient may become critical. Hence, our implementation allows a more frequent use of perturbations in the second stage.

The effectiveness and efficiency of the new approach are clearly shown from the results of solving a large variety of problems. The results on some mathematical test problems can be found in [8], [9] and [10].

IV. A TWO-SECTION TRANSMISSION-LINE TRANSFORMER EXAMPLE

Consider the classical two-section 10:1 transmission-line transformer shown in Fig. 1. Originally proposed by Bandler and Macdonald [11], this problem has been widely used to test minimax algorithms. The error functions (f_j) are given by the reflection coefficient sampled at 11 frequencies normalized with respect to 1GHz: {0.5, 0.6, ..., 1.5}. Madsen and Schjaer-Jacobsen [12] have shown that when we take the characteristic impedances Z_1 and Z_2 as variables and keep the lengths ℓ_1 and ℓ_2 constant at their optimal values (the quarter wavelength at the center frequency), the minimax problem is singular. Fig. 2 shows the minimax contours and illustrates the solution obtained using exact derivatives. If the derivatives were to be estimated by perturbations, 24 function evaluations would have to be performed. Using our gradient approximation, we obtained the solution, as shown in Fig. 3, after 18 function evaluations.

For the same transformer, we also formulate an ℓ_1 problem. The reflection coefficient at the minimax optimum was taken as a measurement from which we attempt to identify the values of Z_1 and Z_2. The solutions obtained with the gradients estimated entirely by perturbations and by our new algorithm are illustrated in Figs. 4 and 5, respectively.

A comparison between Figs. 2 to 5 reveals that the solutions obtained using approximate gradients require more iterations of the optimization but overall fewer function evaluations, which is expected.

V. FET MODELING USING ℓ_1 OPTIMIZATION WITH APPROXIMATE GRADIENTS

A. Introductory Remarks

The use of ℓ_1 optimization, based on its theoretical properties, has been recommended for nonlinear data-fitting and device modeling [1], [7], [13]. Jansen and Koster [14] have investigated the use of generalized ℓ_p optimization in the modeling of microwave transistors, and they concluded that values of p around unity would lead to relatively stable solutions with good convergence properties. A novel approach to robust modeling of microwave devices has been presented by the authors [7] which exploits the unique properties of the ℓ_1 norm and employs the concept of simultaneous processing of multiple circuits. It has the advantage of establishing not only a good equivalent circuit model but also a reliable measure of the self-consistency of the model. In the context of this paper, an example of FET modeling is given to illustrate the ℓ_1 optimization with integrated gradient approximations.

One of the concerns in practical modeling of FET devices is the uniqueness of the solution. A family of solutions may exist which all exhibit a reasonable match between the calculated and measured responses. The approach described in [7] is intended to improve the chance of unique identification of the model parameters by processing simultaneously multiple circuits. In the case of FET modeling, we create multiple circuits by taking measurements on the scattering parameters under several different biasing conditions. From the physical characteristics of the device we know that with respect to different biasing conditions some model parameters should remain almost unchanged while the others should vary smoothly. Therefore, from a family of possible solutions we give preference to the one that exhibits the desired consistency. Such a self-consistent model can be achieved automatically by using the ℓ_1 optimization and choosing those model parameters that are insensitive to bias as common variables.

B. The Model and the Measurements

The small signal equivalent circuit model for the FET is shown in Fig. 6 which is widely used by commercial programs such as TOUCHSTONE [15] and SUPER-COMPACT [16]. The model has 11 parameters that we will consider as optimization variables:

$$\{R_g, R_d, L_s, \tau, R_{ds}, R_i, R_s, C_{gs}, C_{dg}, C_{ds}, g_m\}.$$

The first four parameters are considered to be bias insensitive.

Three sets of 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 were made available by R.A. Pucel [17].

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

C. Formulation of the Problem

Microwave device modeling utilizing multiple circuits has been formulated in general as an ℓ_1 optimization problem by the authors [7]. The following are formulas (12) to (14) in [2]:

$$\underset{x}{\text{minimize}} \sum_{t=1}^{n_c} \sum_{i=1}^{k_t} |f_i^t|, \tag{14}$$

where

$$f_i^t = w_i^t \, [F_i^c(x^t) - (F_i^m)^t] \tag{15}$$

and

$$x = \begin{bmatrix} x^1 \\ x_a^2 \\ \circ \\ \circ \\ x_a^{n_c} \end{bmatrix}, \tag{16}$$

with superscript and index t identifying the t-th circuit. n_c is the number of circuits and k_t is the number of functions arising from the t-th circuit. x^t represents the vector of parameters of the t-th circuit. Vectors x_a^t, $t = 1, ..., n_c$, contain only those parameters that vary between different circuits. They do not include the common variables, i.e., those parameters that assume the same values for all circuits. For each circuit, we combine the common variables and x_a^t to form the vector x^t.

For the FET modeling problem under consideration, which has three sets of measurements, we specialize the formulas as follows:

$$\underset{x}{\text{minimize}} \sum_{t=1}^{3} \sum_{i=1}^{17} \sum_{j=1}^{2} \sum_{k=1}^{2} \{|\mathrm{Re}[f_{jk}^{t}(\omega_i)]| + |\mathrm{Im}[f_{jk}^{t}(\omega_i)]|\}, \qquad (17)$$

where

$$f_{jk}^{t}(\omega_i) = F_{jk}^{t}(x^t,\omega_i) - S_{jk}^{t}(\omega_i). \qquad (18)$$

In (18), f_{jk}^{t} and S_{jk}^{t} are the calculated and measured scattering parameters, respectively, with superscript identifying three different biasing conditions. Having 17 frequency points with real and imaginary parts of the complex S-parameters being treated separately, we have a total of 408 error functions. The variables to be optimized in (17) are defined as

$$x = \begin{bmatrix} x^1 \\ x_a^2 \\ x_a^3 \end{bmatrix} . \qquad (19)$$

The vector x^1 actually has two parts as $x^1 = [x^c \; x_a^1]^T$, where x^c consists of the common variables as

$$x^c = [R_g \; R_d \; L_s \; \tau]^T. \qquad (20)$$

These are the parameters we expect not to change with respect to different bias. The vector x_a^t contains the remaining parameters of model t, namely

$$x_a^t = [R_{ds}^t \; R_i^t \; R_s^t \; C_{gs}^t \; C_{dg}^t \; C_{ds}^t \; g_m^t]^T. \qquad (21)$$

The total number of variables is 25.

D. Results

To solve the problem we have formulated, the ℓ_1 optimizer described in [2] was employed. The gradient required was provided by the approach proposed in this paper. We should point out that in this case the evaluation of exact sensitivities is actually possible using the scheme outlined in [7]. However, it involves lengthy and complicated programming. First of all, two adjoint solutions are needed to evaluate the sensitivity expressions for the admittance matrix. From these expressions the sensitivities of the S-parameters are derived. Since multiple circuits are processed simultaneously, a complex coding scheme is needed to associate functions arising from different circuits with the appropriate variables. It is then very difficult to modify the software when needed. Comparatively, the calculation of the function values alone requires much simpler effort. This, from the view point of reducing software complexity, justifies the pursuit of gradient approximation.

Three experiments were conducted which have used different schemes for gradient approximation. From the starting point given in Table I, they have reached practically the same solution, which is also given in Table I. The match between the calculated and measured responses for the first circuit, at both the starting point and the solution, are shown in Figs. 7 and 8. The match for the other two biasing conditions is similar and hence omitted.

The first experiment corresponds to the conventional approach, in which the gradients were estimated solely by perturbations. A total of 468 circuit simulations were required to reach the solution.

In the second case, the Broyden update without weights was used. Regular corrections were also provided by perturbations for every five iterations. Only 128 circuit simulations were needed for this solution.

For the third experiment, we took advantage of an inherent decomposition in the multi-circuit formulation. Notice that the responses (and error functions) of one circuit are absolutely uncorrelated to the independent parameters (x_a^t) of any other circuits. Obviously, the derivatives corresponding to such decoupled functions and variables are always equal to zero. However, when we use the Broyden update without weights, these derivatives may be changed to some non-zero values, thus introducing apparent errors to the approximation. We can avoid this by using the weighted update. By assigning zero weights to decoupled functions and variables, we can keep the zero derivatives undisturbed throughout the optimization process. The application of this concept has reduced the use of perturbations and led to the solution after only 79 circuit simulations. This represents less than 1/5 of the simulations required by the first experiment as well as a 38% saving in computational effort as compared to the second experiment.

VI. WORST-CASE DESIGN OF A MICROWAVE AMPLIFIER

Worst-case design using optimization techniques in general has been discussed in [18]. Consider a vector of nominal designable parameters

$$\boldsymbol{\phi}^0 = [\phi_1^0 \dots \phi_n^0]^T, \tag{22}$$

a vector of associated tolerances

$$\boldsymbol{\epsilon} = [\epsilon_1 \dots \epsilon_n]^T, \tag{23}$$

and a tolerance region defined by

$$R_\epsilon = \{\boldsymbol{\phi} \mid \boldsymbol{\phi}^0 - \boldsymbol{\epsilon} \leq \boldsymbol{\phi} \leq \boldsymbol{\phi}^0 + \boldsymbol{\epsilon}\}. \tag{24}$$

We seek an optimally centered design such that the specifications are satisfied over the tolerance region. It can be formulated as a minimax problem, as

$$\underset{\boldsymbol{\phi}^0}{\text{minimize}} \; \max_{j} \; \max_{\boldsymbol{\phi} \in R_\epsilon} \; \{f_j(\boldsymbol{\phi})\}, \tag{25}$$

where f_j, $j = 1, \dots, m$, are a set of error functions derived from the design specifications. In practice, we usually consider as candidates for the worst case the vertices of the tolerance region defined by

$$R_v = \{\boldsymbol{\phi} \mid \phi_i = \phi_i^0 + \epsilon_i \, \mu_i, \; \mu_i \in \{-1, 1\}, \; i = 1, \dots, n\}. \tag{26}$$

Consider the worst-case fixed tolerance design of a microwave amplifier. As shown in Fig. 9, the amplifier consists of a NEC70000 FET and five transmission-lines [15]. The FET is characterized by tabulated scattering parameters provided by the manufacturer. The design variables are the characteristic impedance Z and the lengths ℓ_i of the transmission-lines. For each length ℓ_i we assume a five percent tolerance. The design specifications are given by

$$7.05\text{dB} \le 20\log|S_{21}| \le 8.2\text{dB}, \quad \text{for } \omega_j = 6, 7, ..., 18\text{GHz}.$$

A total of 26 error functions (f_j) arise from the upper and lower specifications at 13 frequencies.

The worst-case design was accomplished by two phases of optimization. In the first one, we predicted an initial set of worst-case vertices by first-order changes. For each f_j, a vertex $\boldsymbol{\phi}^j$ was selected by

$$\phi_i^j = \phi_i^0 + \mu_i^j \epsilon_i, \quad \mu_i^j = \text{sign}(\partial f_j/\partial\phi_i), \quad i = 1, ..., n, \tag{27}$$

where the derivatives $\partial f_j/\partial\phi_i$ were estimated at the nominal point at the start of the optimization by perturbations. Consequently, 26 vertices (one for each f_j) were considered for the minimax problem

$$\underset{\boldsymbol{\phi}^0}{\text{minimize}} \ \max_j \ \{f_j(\boldsymbol{\phi}^j)\}. \tag{28}$$

At the solution, by using (27) with respect to the new nominal point, we found that 10 of the worst-case vertices had changed (i.e., the signs of some $\partial f_j/\partial\phi_i$ had changed). The new vertices were added to the worst-case set. The corresponding old vertices were kept, instead of replaced, in order to stabilize the algorithm. We had, therefore, a total of 36 worst-case vertices. A second optimization was performed and at its solution the worst-case set was found to be complete (i.e., no more sign change in (27)).

The nominal parameter values at the starting point and the final solution are given in Table II. The total number of function evaluations is 280, opposed to 585 required if perturbations were used throughout the optimization. Fig. 10 depicts the worst-case envelope at the solution.

VII. PRACTICAL DESIGN OF A 5-CHANNEL MULTIPLEXER

A. Introductory Remarks

A minimax solution of a 5-channel 11GHz noncontiguous band multiplexer was given in detail by Bandler et al. [1]. In order to provide the exact sensitivities required, the theory due to Bandler et al. [19] was implemented in a computer program which has taken months of effort to develop and test. Furthermore, because the sensitivity expressions depend highly on the circuit structure and vary from component to component, every change to the problem, such as assigning different variables, requires expert modification to the software. In fact, the sensitivities with respect to all possible variables were computed even though some of them have not been actually used, otherwise the coding scheme would have become unmanageable. Large amounts of computer memory were required to store various adjoint solutions and intermediate expressions. By utilizing our gradient approximation, it is possible to efficiently design a multiplexer without all these troubles associated with computing the exact sensitivities. The complexity and size of the program can therefore be considerably reduced.

The 5-channel multiplexer provides an excellent illustration of efficient gradient approximations for two reasons. First, it involves 75 variables and, therefore, to rely on perturbations would be prohibitively expensive. To be more specific, suppose that we use the initial parameter values and specifications suggested by Bandler et al. [1]. The multiplexer responses at the starting point are shown in Fig. 11. We have reached a result similar to the one reported in [1] after 50 iterations of optimization using exact derivatives. To rely on perturbations for the gradients, we would have to compute multiplexer responses 3800 times (50 × 76). We will show that efficient gradient approximations reduce the number of response evaluations significantly.

Also, this example is naturally suited for the use of the weighted Broyden update described earlier in this paper. From Fig. 11 it is intuitively obvious that the response functions at lower frequencies should be almost independent of the variables that are related to the filters of channels 1 and 2 (channel 1 has the highest center frequency). Similarly, the responses at higher frequencies are almost independent of the variables related to the filters of channels 3, 4 and 5. We will demonstrate the advantage of using the weighted update.

B. Results

Details of the 5-channel multiplexer structure, such as the channel center frequencies, bandwidths and coupling matrices, can be found in [1]. The channel filters are assumed lossy. Frequency dispersion and nonideal junctions are also taken into account. For all the results that follow, we have used the same specifications and starting point as given in [1]. Three experiments were performed each using a different method for gradient approximation.

In the first experiment, perturbations were used only at the starting point but not during the optimization. The approximation of gradients relied on the Broyden update in conjunction with the special iterations, which was similar to the methods of Madsen [5] and Zuberek [6]. The optimization stopped after 266 response evaluations, of which 75 were used for the initial perturbations. The responses at this solution as depicted in Fig. 12 are considerably inferior to the result reported in [1]. The optimization has stopped prematurely. This experiment has demonstrated that the Broyden update may not be sufficient for large-scale nonlinear problems.

In a second experiment, regular corrections were provided during the optimization by perturbations for every 20 iterations. After 500 response evaluations, of which 375 were used for perturbations, we obtained the responses shown in Fig. 13. Continuing the process for another 500 response evaluations the responses shown in Fig. 14 were achieved, which are as good as the ones in [1]. From the starting point, a total of 1000 response evaluations was performed. Recall that 3800 response evaluations would be required if the gradient calculations were simply replaced by perturbations.

The third experiment is intended to demonstrate the weighted update proposed in this paper. To apply this updating formula, a weight w_{ji} is set to zero if we know that a function f_j is almost independent of a variable x_i. For instance, the insertion loss of channels 3, 4 and 5 and the common port return loss over the passbands of these channels are almost independent of the filter couplings in channels 1 and 2. Similarly, the responses within the frequencies of channels 1 and 2 are almost independent of the filter couplings in channels 3, 4 and 5. Therefore, we set the corresponding weights to zero.

Utilizing the weighted update, we optimize the multiplexer without any regular correction by perturbations. After 500 response evaluations we obtained the responses shown in Fig. 15. By comparing this result with experiment 1 we can clearly see that the use of appropriate weights has prevented the optimization from stopping prematurely. We can also conclude from a comparison between experiments 2 and 3 (also, between Figs. 13 and 15) that the application of the weighted update has effectively reduced the use of time-consuming perturbations.

VIII. CONCLUSIONS

A new algorithm for gradient approximations has been presented. Integration of this algorithm with powerful gradient-based optimization techniques has been described and illustrated by the minimax and ℓ_1 implementations. The effectiveness and efficiency of the proposed approach has been demonstrated through diverse examples of practical significance, including FET modeling, worst-case centering and multiplexer design. A weighted update has also been proposed which exploits possible sparsity and decoupled structures to further reduce the computations involved in estimating gradients. The new approach is very useful when analytical evaluation of partial derivatives is unavailable or tedious. The prospect of integrating our method with existing CAD packages and thus bringing the full power of advanced optimization techniques into practical microwave applications is especially promising.

ACKNOWLEDGEMENTS

Thanks are extended to Dr. R.A. Pucel of Raytheon Company, Research Division, Lexington, MA, for his permission to use the FET data quoted in this paper. The first author must also acknowledge helpful discussions with Dr. Pucel on the subject of device modeling for CAD. The useful comments offered by the reviewers are also appreciated, including bringing reference [14] to the authors' attention.

REFERENCES

[1] 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.

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

[3] C.G. Broyden, "A class of methods for solving nonlinear simultaneous equations", *Mathematics of Computation*, vol. 19, 1965, pp. 577-593.

[4] 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.

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

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

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

[8] 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.

[9] J.W. Bandler, S.H. Chen, S. Daijavad and K. Madsen, "Efficient optimization with integrated gradient approximations, Part I: algorithms", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-86-12, 1986.

[10] J.W. Bandler, S.H. Chen, S. Daijavad and K. Madsen, "Efficient optimization with integrated gradient approximations, Part II: implementation", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-86-13, 1986.

[11] J.W. Bandler and P.A. Macdonald, "Cascaded noncommensurate transmission-line networks as optimization problems", *IEEE Trans. Circuit Theory*, vol. CT-16, pp. 391-394.

[12] K. Madsen and H. Schjaer-Jacobsen, "Singularities in minimax optimization of networks, *IEEE Trans. Circuits and Systems*, vol. CAS-23, 1976, pp. 456-460.

[13] 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.

[14] R.H. Jansen and N. Koster, "Generalized least pth optimization in the modeling of microwave transistors", *AEÜ*, vol. 31, 1977, pp. 475-477.

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

[16] SUPER-COMPACT User's Manual, Communications Consulting Corp., Paterson, NJ 07504, 1986.

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

[18] 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.

[19] 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.

APPENDIX

FORMULAS FOR POWELL'S SPECIAL ITERATIONS

The formulas for computing the increment vector for a special iteration, as derived by Powell [4], are as follows.

An n by n (n being the dimension of x) orthogonal matrix $\mathbf{D}_k$ is constructed at each iteration. Denote the rows of $\mathbf{D}_k$ by $\mathbf{d}_i^T$, i = 1, 2,..., n. At a special iteration, the increment vector is set to a multiple of the first row vector of $\mathbf{D}_k$, as

$$\mathbf{h}_k = \Lambda_k \, \mathbf{d}_1, \tag{A.1}$$

where Λ_k is a parameter controlling the step size of $\mathbf{h}_k$. Usually it is set to the step size of the latest ordinary iteration.

At the starting point $\mathbf{D}_1$ is set to an identity matrix. At the kth iteration. $\mathbf{D}_k$ is revised to produce $\mathbf{D}_{k+1}$. We use y_i^T for the rows of $\mathbf{D}_{k+1}$. For a special iteration, we simply let

$$\begin{gathered} \mathbf{y}_i = \mathbf{d}_{i+1}, \quad i = 1, 2, \ldots, n-1, \\ \mathbf{y}_n = \mathbf{d}_1. \end{gathered} \tag{A.2}$$

For an ordinary iteration, the following steps take place.

Step 1 Compute $\sigma_i = \mathbf{d}_i^T \mathbf{h}_k$, i = 1, 2,..., n.

Step 2 Find t which is the greatest integer such that $\sigma_t \neq 0$.

Step 3 Let $\alpha_t = 0$ and $\mathbf{z}_t = 0$. For i = t-1, t-2, ..., 1, compute

$$\begin{gathered} \mathbf{z}_i = \mathbf{z}_{i+1} + \sigma_{i+1}\mathbf{d}_{i+1}, \\ \alpha_i = \alpha_{i+1} + \sigma_{i+1}^2, \\ \mathbf{y}_i = (\alpha_i \mathbf{d}_i - \sigma_i \mathbf{z}_i)/[\alpha_i(\alpha_i + \sigma_i^2)]^{\frac{1}{2}}. \end{gathered} \tag{A.3}$$

Step 4 Let $\mathbf{y}_i = \mathbf{d}_{i+1}$, i = t, t+1,..., n-1. Let $\mathbf{y}_n = \mathbf{h}_k/(\mathbf{h}_k^T \mathbf{h}_k)^{\frac{1}{2}}$.

TABLE I

PARAMETER VALUES OF THE FET MODELS

Parameter	Starting Point	Solution		
		Case 1	Case 2	Case 3
R_g (OH)	1.0	2.6025	2.6025	2.6025
R_d (OH)	1.0	3.7630	3.7630	3.7630
R_{ds} (KOH)	0.143	0.1992	0.1638	0.1632
R_i (OH)	1.0	0.0099	0.0999	0.3891
R_s (OH)	1.0	1.0016	0.9220	0.6482
L_s (nH)	0.02	0.0039	0.0039	0.0039
C_{gs} (pF)	1.4	0.7181	0.4417	0.3454
C_{dg} (pF)	0.07	0.0306	0.0475	0.0609
C_{ds} (pF)	0.4	0.2228	0.2229	0.2151
g_m (/OH)	0.09	0.0696	0.0521	0.0410
τ (ps)	7.0	3.9558	3.9558	3.9558

Biasing Conditions

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

The starting points for the three circuits are identical.

TABLE II

PARAMETER VALUES OF THE MICROWAVE AMPLIFIER

Parameter	Starting Point	Solution
ℓ_1	52.96	69.01
ℓ_2	148.13	152.01
ℓ_3	26.80	18.48
ℓ_4	24.01	5.10
ℓ_5	46.63	36.49
Z	81.27	126.39

The starting point is a minimax nominal design

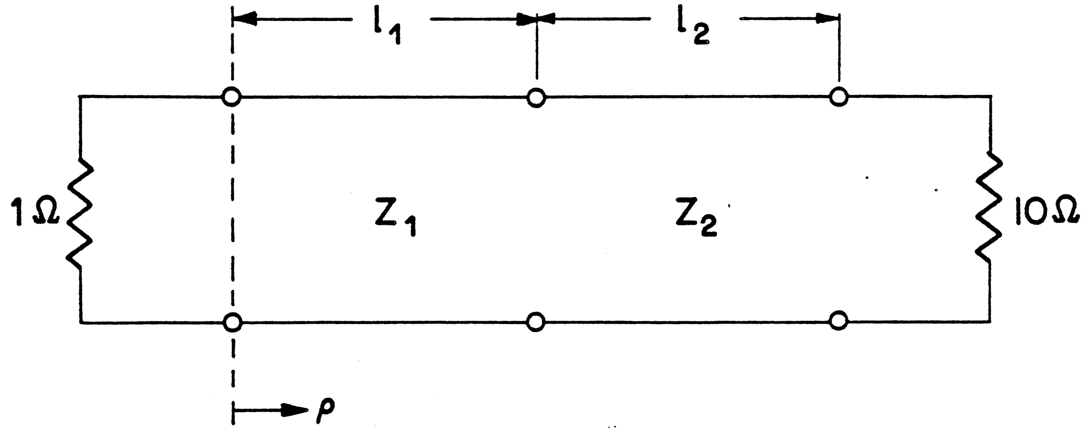


Fig. 1 Two-section, 10:1 transmission-line transformer.

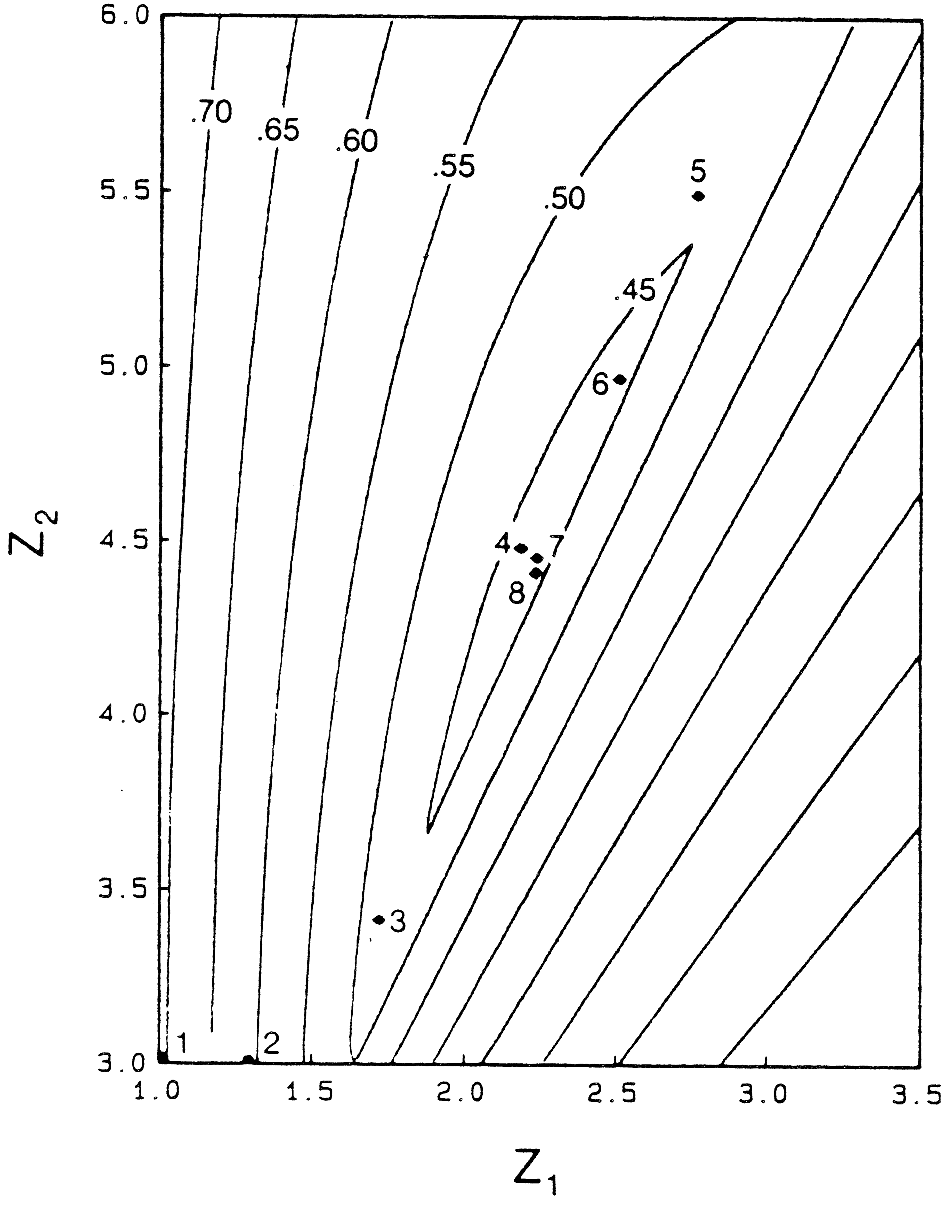


Fig. 2 Minimax contours for the two-dimensional singular minimax problem arising from optimization of the two-section transmission-line transformer. Eight iterations using exact gradients are illustrated.

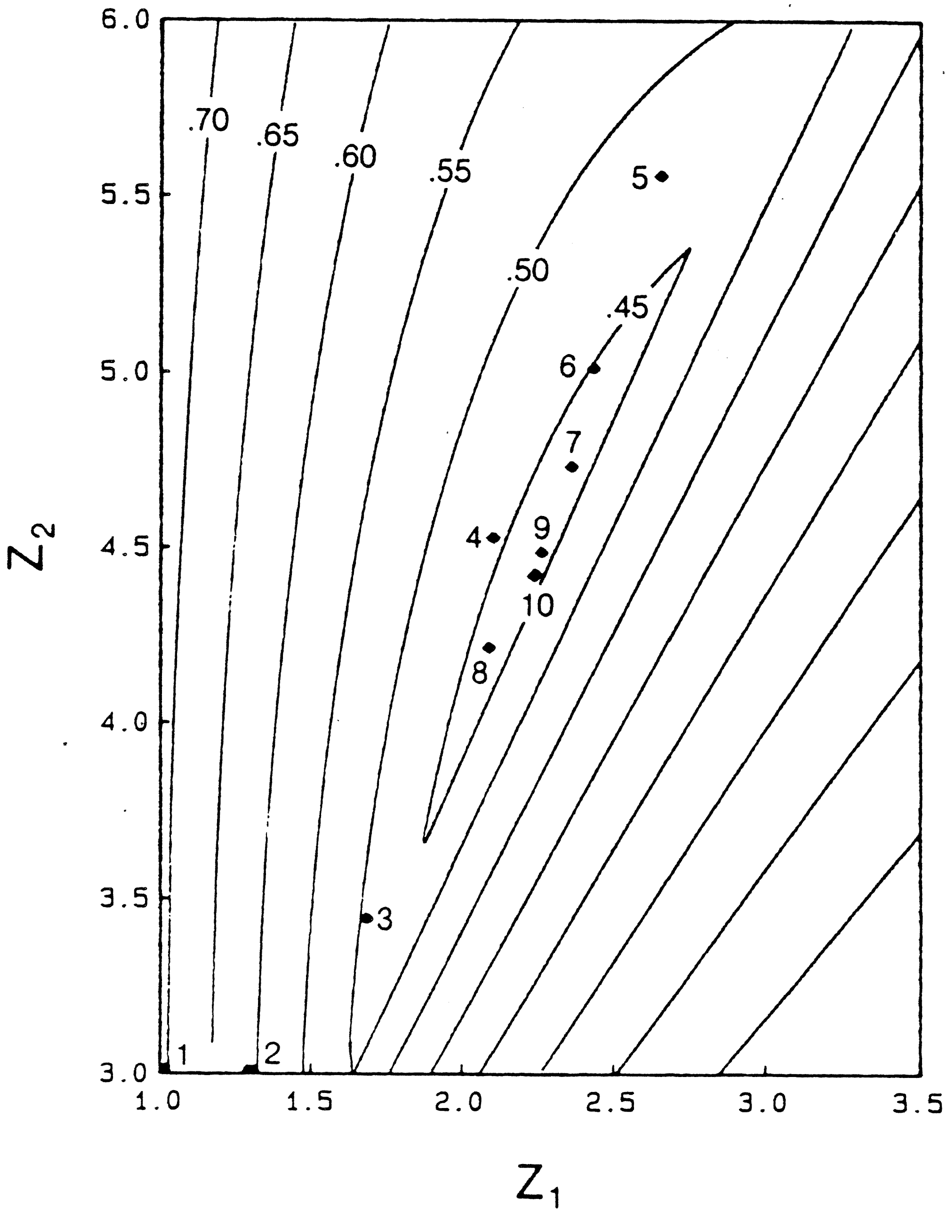


Fig. 3 Minimax optimization of the two-section transmission-line transformer. Ten iterations using approximate gradients are illustrated.

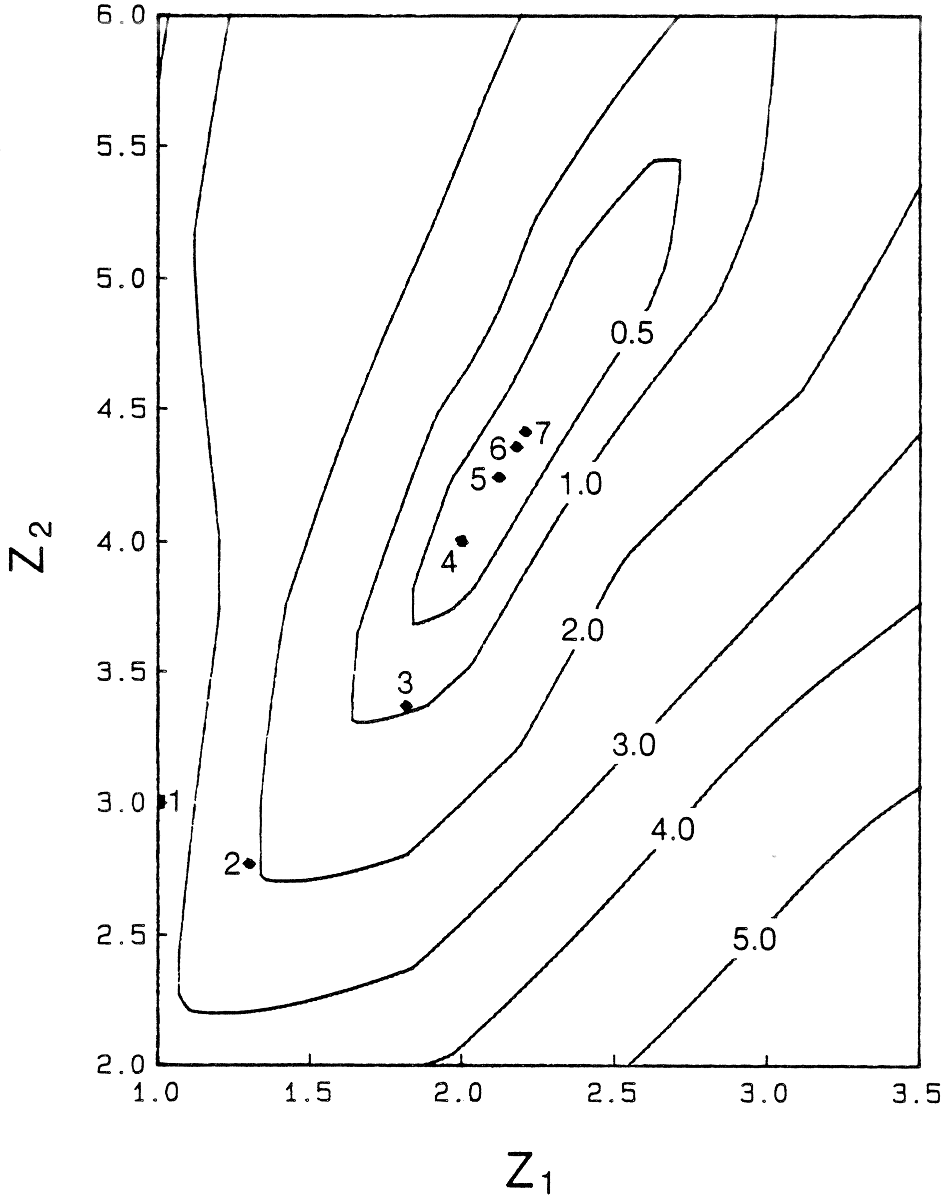


Fig. 4 ℓ_1 contours for problem arising from parameter identification of the two-section transmission-line transformer. Using perturbations for the gradients, the solution required 14 iterations (42 function evaluations). The first 7 iterations are illustrated.

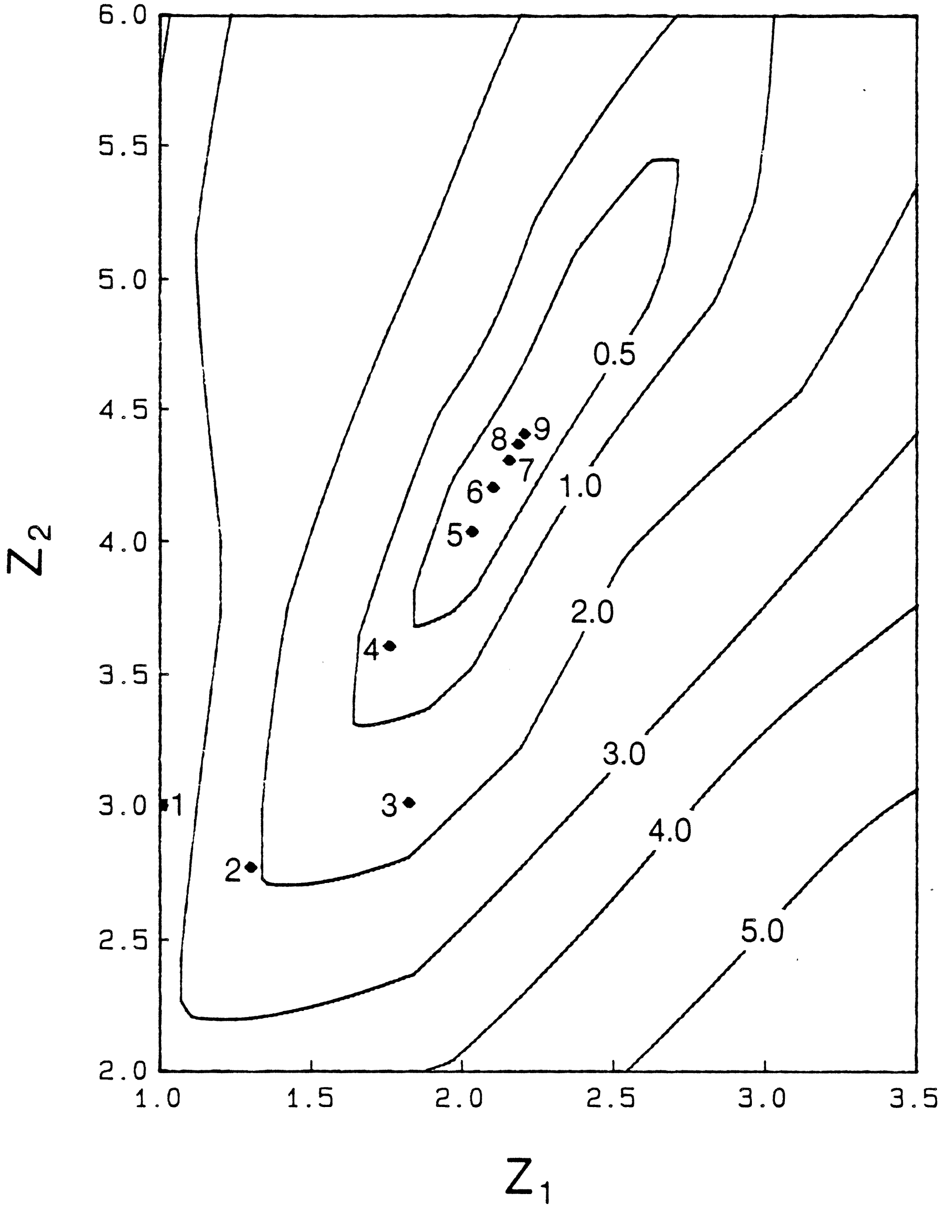


Fig. 5 Parameter identification of the two-section transmission-line transformer using ℓ_1 optimization with approximate gradients. The solution has required 19 iterations and 27 function evaluations. The first 9 iterations are illustrated.

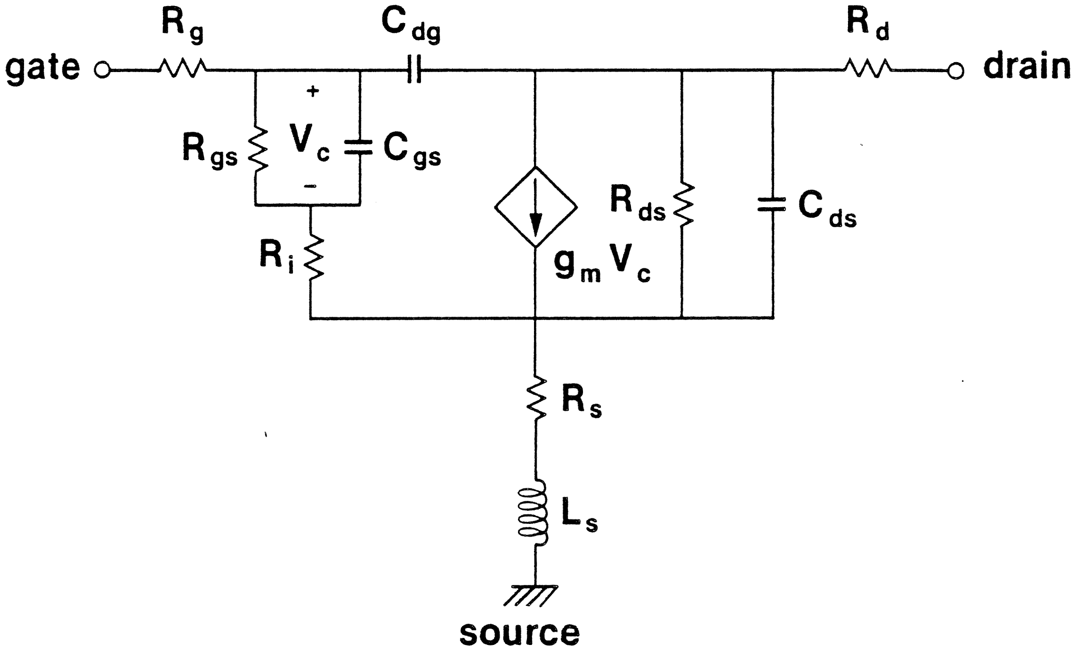


Fig. 6 The small-signal equivalent circuit model for FET devices.

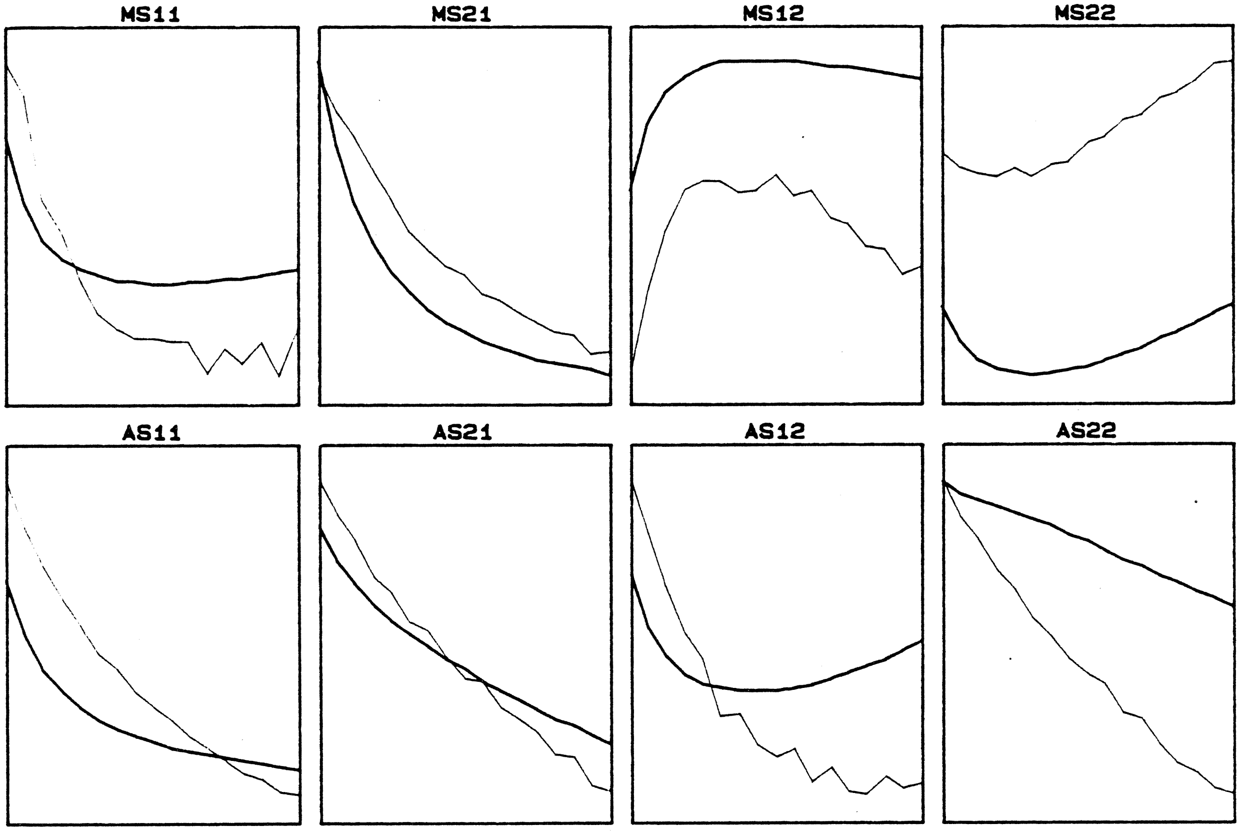


Fig. 7 The scattering parameter match between the FET model and the measurements at the starting point, for V_{ds} = 4V, V_{gs} = 0V and I_{ds} = 177mA.

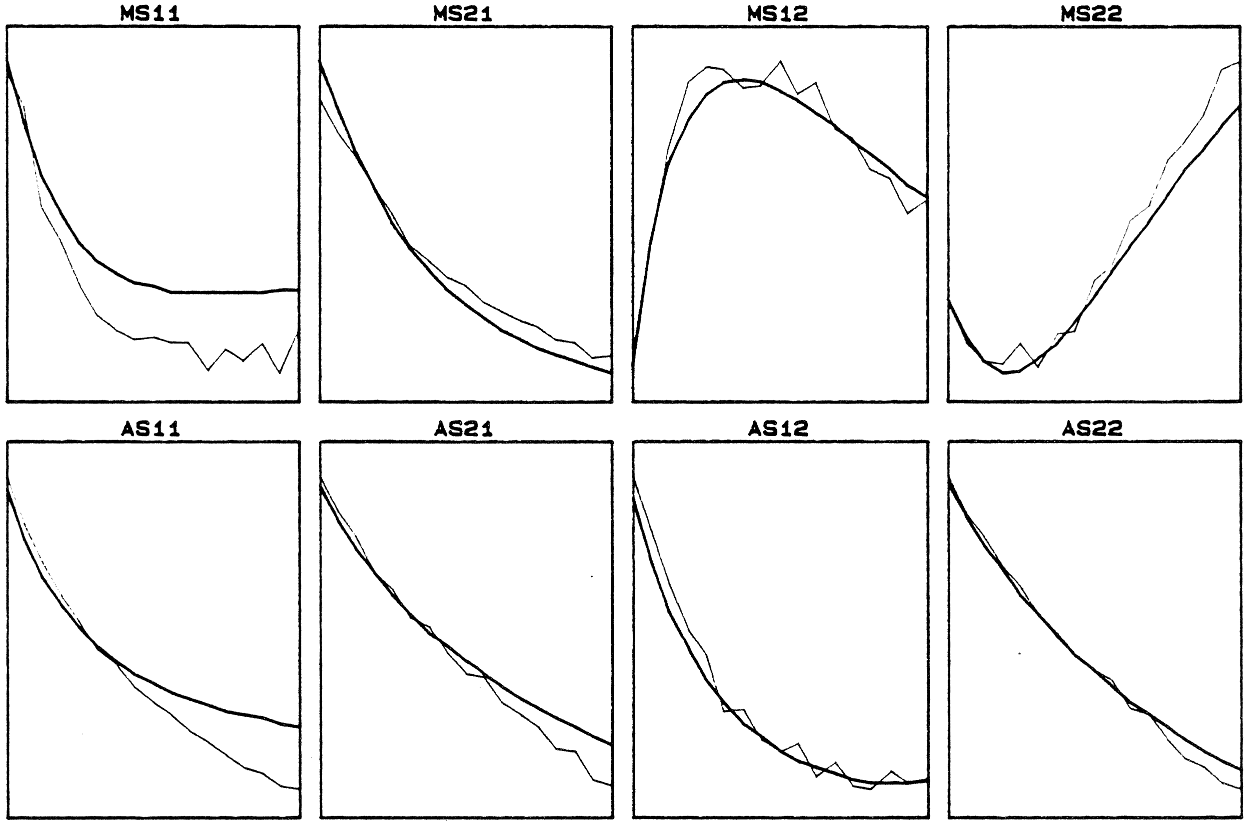


Fig. 8 The scattering parameter match between the FET model and the measurements at the solution, for V_{ds} = 4V, V_{gs} = 0V and I_{ds} = 177mA.

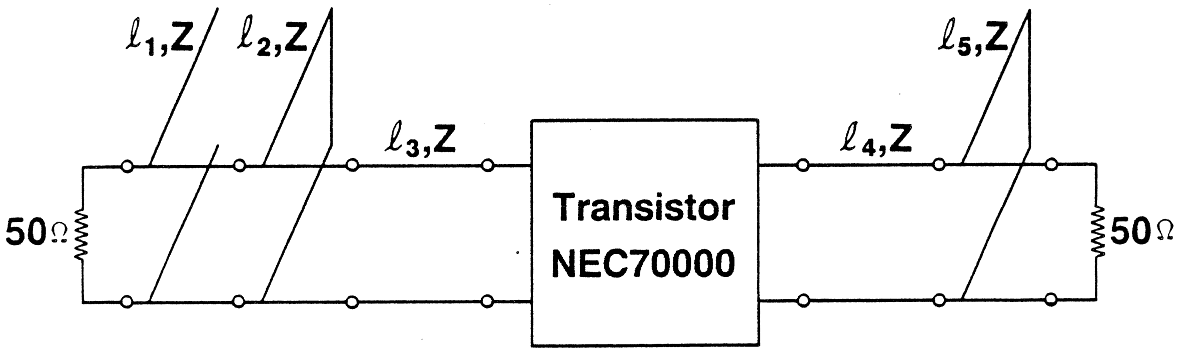


Fig. 9 A microwave amplifier.

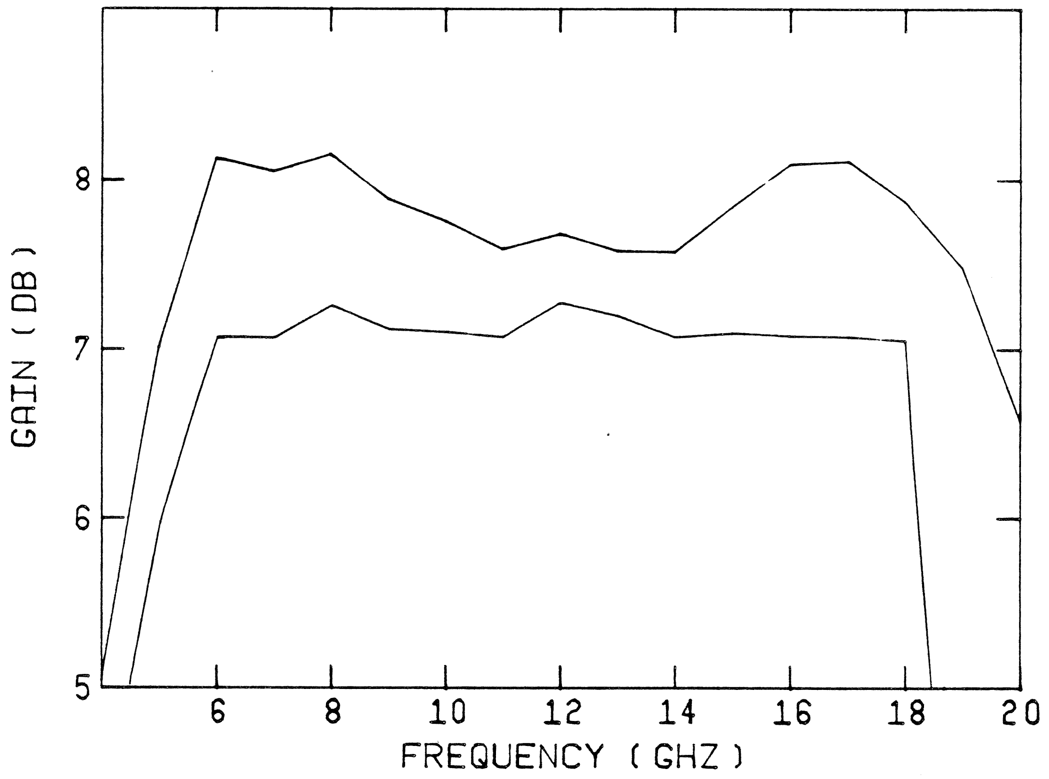


Fig. 10 Worst-case envelope for the amplifier response at the centered solution.

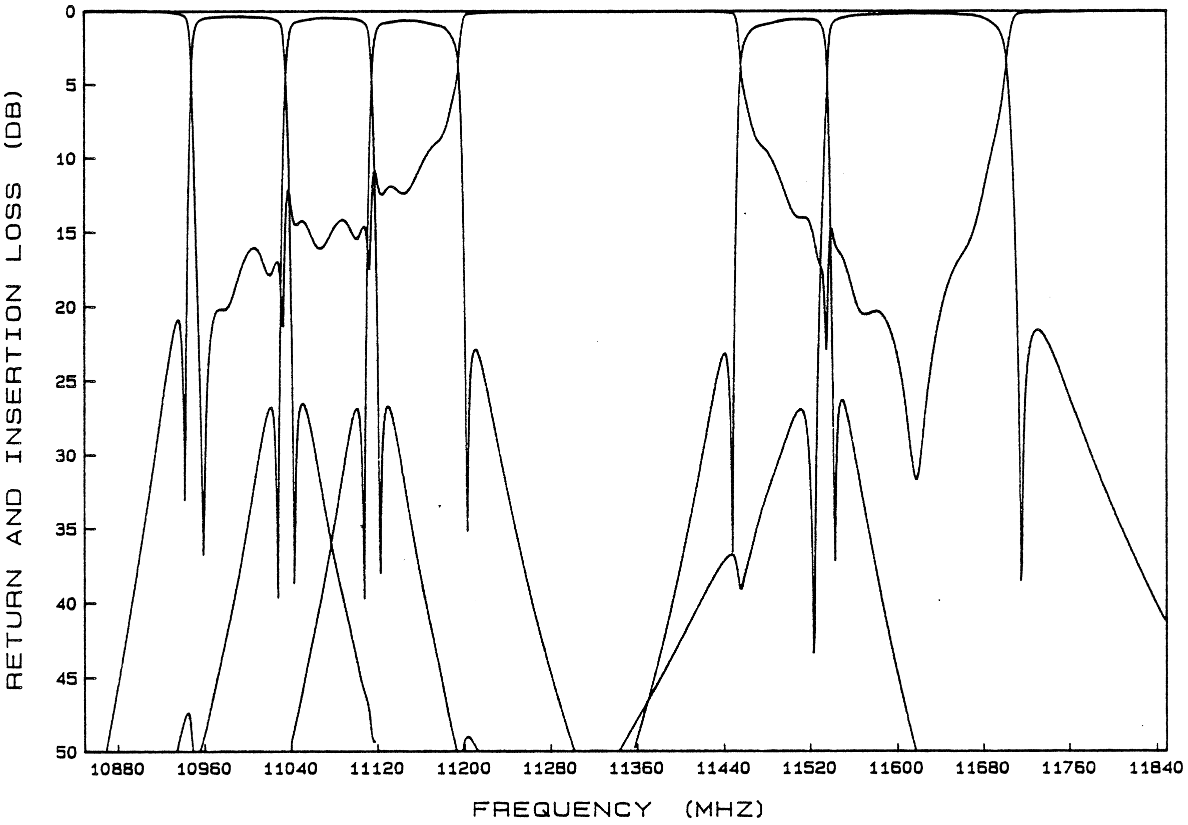


Fig. 11 Responses of the 5-channel multiplexer at the starting point.

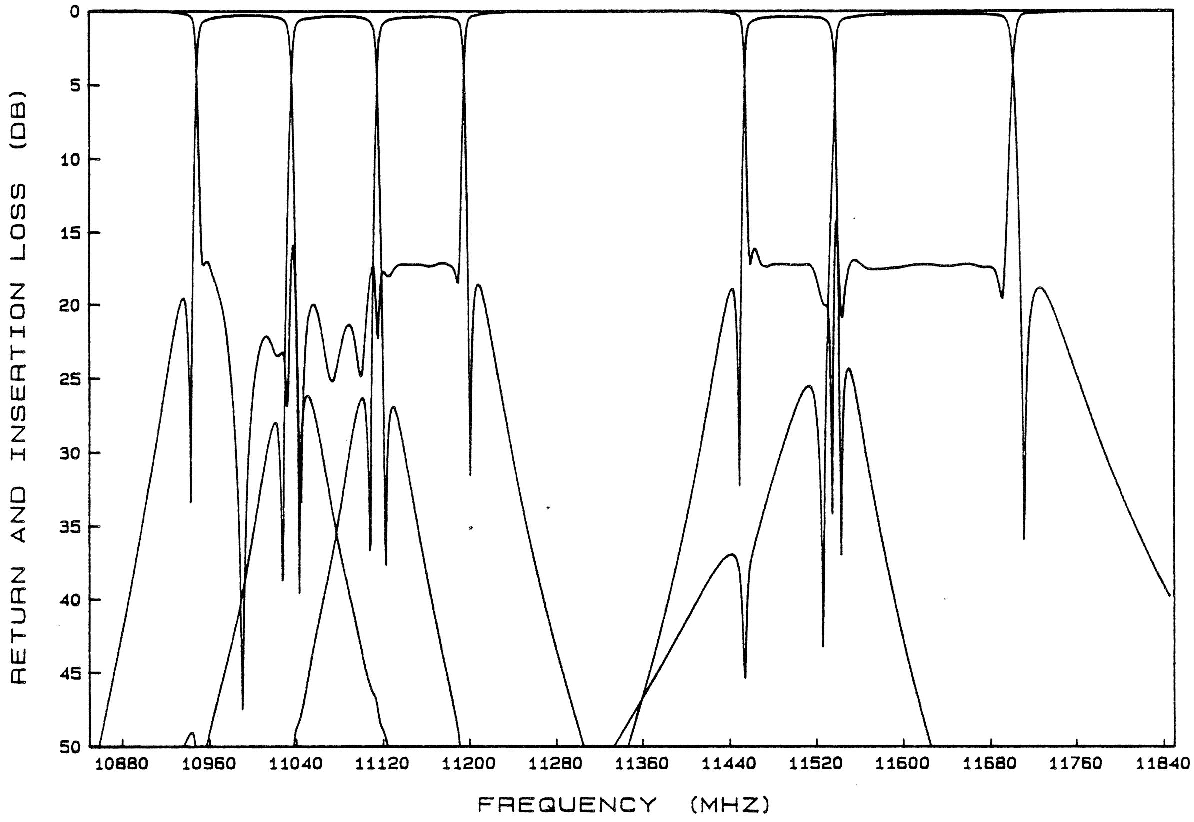


Fig. 12 Responses of the 5-channel multiplexer obtained using only the Broyden update and special iterations for gradient approximations. The optimization has stopped prematurely.

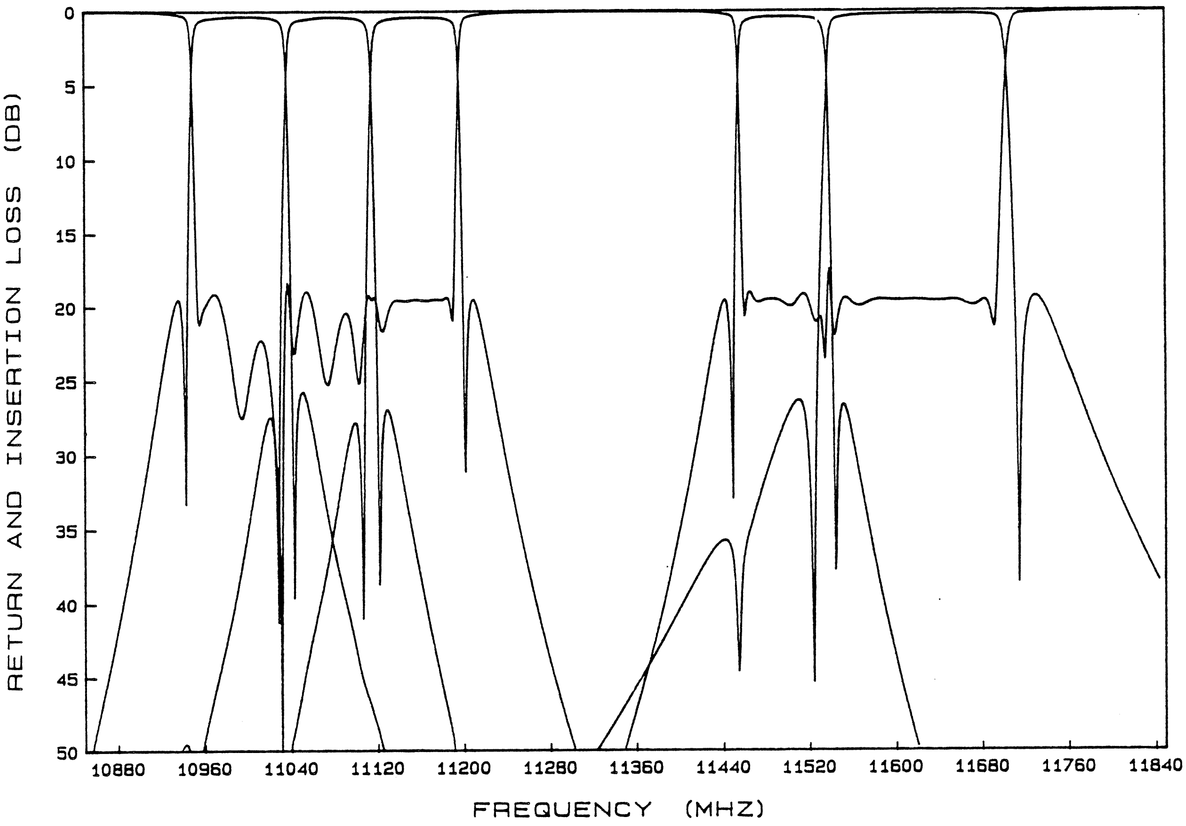


Fig. 13 Responses of the 5-channel multiplexer obtained after 500 response evaluations. Regular corrections to the approximate gradient by perturbations were provided for every 20 iterations.

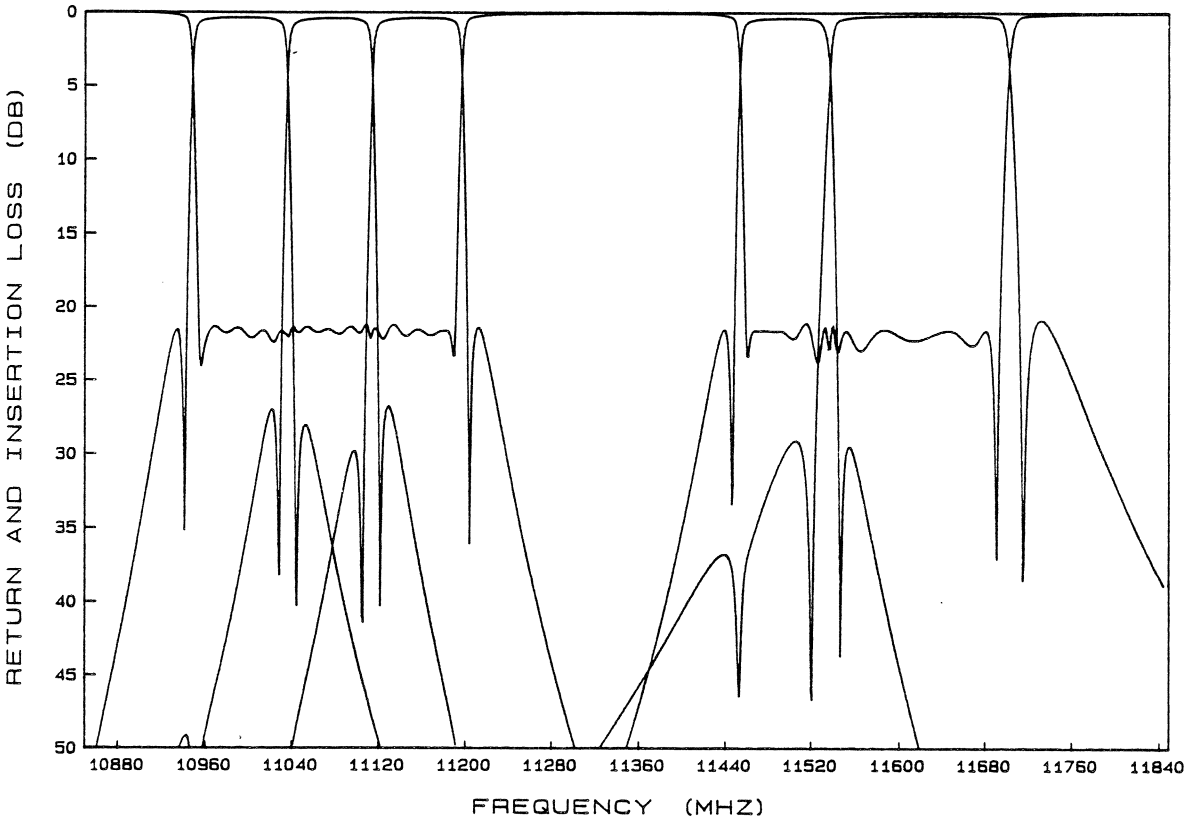


Fig. 14 Responses of the 5-channel multiplexer obtained by continuing the process described in Fig. 13 for another 500 response evaluations.

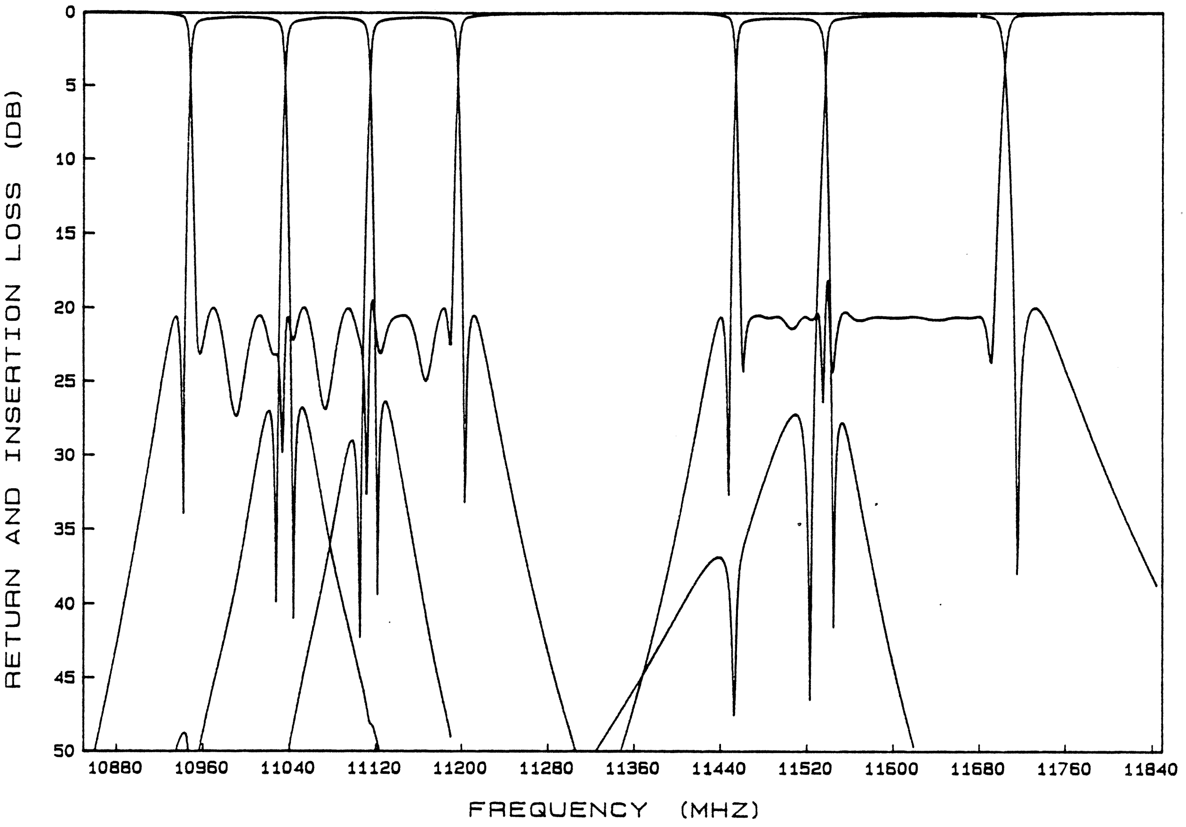


Fig. 15 Responses of the 5-channel multiplexer obtained using the weighted update for gradient approximations. The use of appropriate weights has effectively prevented the optimization from stopping prematurely and reduced the use of time-consuming perturbations.