[8] ____, "Some properties of scattering Hurwitz polynomials," *Arch. Elek. Ubertagung*, vol. 38, pp. 171–176, Mar. 1984.

[9] T. Koga, "Synthesis of finite passive n-ports with prescribed positive real matrices of several variables," *IEEE Trans. Circuit Theory*, vol. CT-15, pp. 2–23, Mar. 1968.

[10] N. K. Bose, *Applied Multidimensional Systems Theory*. New York: Van Nostrand Reinhold, 1982.

[11] A. Fettweis and S. Basu, "New results on stable multidimensional polynomials—Part I: Continuous case," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 1221–1232, Oct. 1987.

[12] A. Fettweis and G. Linnenberg, "An extension of the maximum-modulus principle for applications to multidimensional networks," *Arch. Elek. Ubertragung*, vol. 38, pp. 131–135, Feb. 1984.

[13] D. C. Youla, "The synthesis of networks containing lumped and distributed elements," presented at the Symp. on Generalized Networks, Polytechnic Institute of Brooklyn, NY, pp. 289–343, 1966.

[14] T. Koga, "Synthesis of finite passive n-ports with described two-variable reactance matrices," *IEEE Trans. Circuit Theory*, vol. CT-13, pp. 31–52, 1966.

[15] A. Kummert, "Synthesis of two-dimensional lossless *m*-ports with prescribed scattering matrix," *Circuits, Systems, and Signal Processing*, vol. 8, no. 2, 1989.

Efficient Quadratic Approximation for Statistical Design

RADOSLAW M. BIERNACKI, JOHN W. BANDLER,
JIAN SONG, AND QI-JUN ZHANG

Abstract —**A highly efficient approach to quadratic approximation of circuit responses is presented. Because it uses a fixed pattern of base points, this approach requires extremely small amounts of CPU time and storage space. Using this approach, a major obstacle for the traditional quadratic approximation to deal with large problems, namely, the prohibitive requirement for storage and computational effort, is effectively eliminated. The accuracy and efficiency of this quadratic approximation approach are strongly demonstrated by results of two statistical circuit design examples.**

I. Introduction

In order to make existing statistical circuit design methods more practically useful, many approaches have been devised to reduce very costly computational effort by approximating acceptable regions or circuit responses. Quadratic approximation has proven suitable and successful [1]–[4]. However, the determination of a quadratic model itself for a problem with a larger number of variables may be too expensive.

For a circuit with 50 elements, the number of coefficients in the quadratic model is 1326. The calculation of the coefficients in a traditional manner involves 1326 circuit simulations and solving a linear system of 1326 equations. Besides all the coefficients, the matrix of the linear system requires storage of a 1326 by 1326 array. Determining a quadratic approximation to the response of such a circuit creates quite a large problem in the terms of computer time and storage, although the circuit itself may be of a moderate scale. Therefore for large scale problems the traditional approaches that aim to obtain unique quadratic models do not effectively reduce computational costs.

Biernacki and Styblinski [4] introduced the concept of the maximally flat interpolation and presented an updating algorithm. The most significant property of their approach is that the method allows the number of actual circuit simulation required for an accurate model to be much less than that needed for a full unique quadratic approximation. However, the computational requirement of the method, especially storage space, is still high.

In this paper we substantially enhance the maximally flat quadratic interpolation. Our approach makes use of a fixed pattern of points at which simulation is performed, resulting in very low computational requirements for both CPU time and storage. The basic concept is reviewed in Section II. Our new approach is described in Section III. Section IV compares the efficiency of our approach with that of the original maximally flat quadratic approximation. A brief overview of the design centering approach used is provided in Section V. Two examples of circuit statistical design are given in Section VI. Finally, Section VII contains the conclusions.

II. The Basic Concept of the Maximally Flat Quadratic Approximation

A quadratic model in polynomial form to be used to approximate a given function $f(\boldsymbol{x})$, $\boldsymbol{x}=[x_1\ \ x_2\ \ \cdots\ \ x_n]^T$, can be written as

$$q(\boldsymbol{x}) = a_0 + \sum_{i=1}^{n} a_i(x_i - r_i) + \sum_{\substack{j,i=1\\ j\geqslant i}}^{n} a_{ij}(x_i - r_i)(x_j - r_j) \quad (1)$$

where $\boldsymbol{r}=[r_1\ \ r_2\ \ \cdots\ \ r_n]^T$ is a known reference point. The form of the quadratic function used is similar to that of [4]. However, $q(\boldsymbol{x})$ is defined here w.r.t. the reference point $\boldsymbol{r}$ rather than w.r.t. the origin. Note that the subscript notation is that each coefficient can be easily identified with its corresponding x term, e.g., a_{ij} is the coefficient of $(x_i - r_i)(x_j - r_j)$. Determining a quadratic model is equivalent to determining all its coefficients, which are now unknowns in (1).

Suppose that m $(m > n+1)$ evaluations of $f(\boldsymbol{x})$ are performed at some points $\boldsymbol{x}^i$, $i=1,2,\cdots,m$. These points are called the base points. Using $f(\boldsymbol{x}^i)$, we set up a system of linear equations

$$\begin{bmatrix} \boldsymbol{Q}_{11} & \boldsymbol{Q}_{12} \\ \boldsymbol{Q}_{21} & \boldsymbol{Q}_{22} \end{bmatrix} \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{v} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_1 \\ \boldsymbol{f}_2 \end{bmatrix} \quad (2)$$

where $\boldsymbol{a}$ and $\boldsymbol{v}$ are arranged to have the following orders: $\boldsymbol{a}=[a_0\ a_1\ a_2\ \cdots\ a_n]^T$ and $\boldsymbol{v}=[a_{11}\ a_{22}\ \cdots\ a_{nn} a_{12} a_{13}\ \cdots\ a_{n-1,n}]^T$, respectively. The vectors $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ are of dimensions $n+1$ and $m-(n+1)$, respectively. They contain function values $f(\boldsymbol{x}^i)$. The matrix $\boldsymbol{Q}_{ij}$, $i, j=1,2$, is determined by the coordinates of the base points and of $\boldsymbol{r}$.

Similarly to [4], the reduced system with variables $\boldsymbol{v}$ is obtained as

$$\boldsymbol{C}\boldsymbol{v} = \boldsymbol{e} \quad (3)$$

where

$$\boldsymbol{C} = \boldsymbol{Q}_{22} - \boldsymbol{Q}_{21}\boldsymbol{Q}_{11}^{-1}\boldsymbol{Q}_{12} \quad (4)$$

Manuscript received July 21, 1988; revised January 25, 1989. This work was supported in part by Optimization Systems Associates Inc., and in part by the Natural Sciences and Engineering Council of Canada under Grant A7239. This paper was recommended by Associate Editor T. T. Vu.

R. M. Biernacki and J. W. Bandler are with Optimization Systems Associates Inc., Dundas, Ont., Canada L9H 5E7, and also with the Simulation Optimization Systems Research Laboratory and the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ont., Canada L8S 4L7.

J. Song is with the Simulation Optimization Systems Research Laboratory and the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ont., Canada L8S 4L7.

Q. J. Zhang is with Optimization Systems Associates Inc., Dundas, Ont., Canada L9H 5E7.

IEEE Log Number 8929299.

0098-4094/89/1100-1449$01.00 ©1989 IEEE

and

$$e = f_2 - Q_{21}Q_{11}^{-1}f_1. \tag{5}$$

If $m < (n+1)(n+2)/2$, the above system is underdetermined.

When the least squares constraint is applied to v, the unique solution to (3) can be found as

$$v = C^T(CC^T)^{-1}e \tag{6}$$

and a is readily obtained as

$$a = Q_{11}^{-1}f_1 - Q_{11}^{-1}Q_{12}v. \tag{7}$$

Then both v, the minimal Euclidean norm solution of (3), and a give the maximally flat quadratic interpolation in the form of (1) to $f(x)$. The term of the maximally flat quadratic approximation comes from the mechanism of the least squares constraint that forces the second order derivatives to be as small as possible.

III. Approach Using a Fixed Pattern of Base Points

In the original scheme of [4] all base points are randomly selected. This type of selection allows certain freedom. However, several large matrices have to be stored and manipulated. For instance, matrix C needs an array with dimension $n \times (n+1) \times (m-(n+1))/2$. Meanwhile, some fairly involved calculations, such as matrix inversion, or equivalent calculations shown in (4)–(7), are required. Even a circuit of a reasonable size may demand large storage space and CPU time. Here, we shall propose a new approach which is based on a fixed pattern of base points. The regularity of the pattern will greatly reduce storage and simplify the calculation of coefficients.

In our approach, only m $(n+1 < m \leqslant 2n+1)$ base points are used. The reference point r is selected as the first base point x^1. The next n base points are determined by perturbing one variable at a time around r, i.e.,

$$x^{i+1} = r + [0 \cdots 0 \quad \beta_i \quad 0 \cdots 0]^T, \qquad i = 1,2,\cdots,n \tag{8}$$

where β_i is a predetermined perturbation. It can be shown that the first $n+1$ base points lead to very simple forms of matrices Q_{11}^{-1} and $Q_{11}^{-1}Q_{12}$. They are

$$Q_{11}^{-1} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -1/\beta_1 & 1/\beta_1 & & \\ \vdots & & \ddots & \mathbf{0} \\ & \mathbf{0} & & \\ -1/\beta_n & & & 1/\beta_n \end{bmatrix} \tag{9}$$

and

$$Q_{11}^{-1}Q_{12} = \left[\begin{array}{cccc|c} 0 & \cdots & & 0 & \\ \beta_1 & & & \mathbf{0} & \\ & \ddots & & & \mathbf{0} \\ \mathbf{0} & & & \beta_n & \end{array}\right]. \tag{10}$$

Because of this simple pattern they need not be stored in matrix form.

After the first $n+1$ base points, the remaining $m-(n+1)$ points follow to provide the second-order information on the function. Similarly to the base points defined in (8), the consecutive base points are selected by also perturbing one variable at a time. For simplicity, these base points are determined by consecutively perturbing the variables in r, that is

$$x^{n+1+i} = r + [0 \quad 0 \cdots 0 \quad \gamma_i \quad 0 \cdots 0]^T, \qquad i = 1,2,\cdots,k \tag{11}$$

where γ_i is another perturbation of r_i, which must not equal β_i, and k equals $m-(n+1)$. Under this arrangement matrices Q_{21} and Q_{22} have regular structures. Substituting Q_{21} and Q_{22} into (4), the matrix C takes a concise analytical form

$$C = \left[\begin{array}{ccc|c} (\gamma_1-\beta_1)\gamma_1 & & \mathbf{0} & \\ & \ddots & & \\ & (\gamma_i-\beta_i)\gamma_i & & \mathbf{0} \\ & & \ddots & \\ \mathbf{0} & & (\gamma_k-\beta_k)\gamma_k & \end{array}\right] \tag{12}$$

and the vector e can be expressed by

$$e = f_2 - \left[\begin{array}{cccccc|c} 1-\gamma_1/\beta_1 & \gamma_1/\beta_1 & & & & & \\ 1-\gamma_2/\beta_2 & & \gamma_2/\beta_2 & & & & \\ \vdots & & & \ddots & \mathbf{0} & & \\ 1-\gamma_i/\beta_i & & & & \gamma_i/\beta_i & & \mathbf{0} \\ \vdots & \mathbf{0} & & & & \ddots & \\ 1-\gamma_k/\beta_k & & & & & \gamma_k/\beta_k & \end{array}\right] f_1. \tag{13}$$

Substituting (12) and (13) into (6), the coefficients are determined by

$$a_{ii} = \left\{ \left[f(x^{n+1+i}) - f(x^1) \right] \big/ \gamma_i - \left[f(x^{i+1}) - f(x^1) \right] \big/ \beta_i \right\} \Big/ (\gamma_i - \beta_i), \qquad i = 1,2,\cdots,k \tag{14a}$$

$$a_{ii} = 0, \qquad i = k+1,\cdots,n \tag{14b}$$

and

$$a_{ij} = 0, \qquad i \neq j \qquad i,j = 1,2,\cdots,n. \tag{14c}$$

The coefficients a_0 and a_i are easily obtained as

$$|a|_0 = f(x^1) \tag{15a}$$

and

$$a_i = \left[f(x^{i+1}) - f(x^1) \right] / \beta_i - \beta_i a_{ii}, \qquad i = 1,2,\cdots,n. \tag{15b}$$

The maximally flat quadratic interpolation, using a fixed pattern of base points defined here, has an interesting property. All the coefficients of the mixed terms, a_{ij} for $i \neq j$, are conveniently forced to be zero because no related information can be extracted from the fixed pattern. Any of the a_{ii}'s in (14a), $i \leqslant k$, can be nonzero because double perturbations are made along a straight line parallel to the ith axis. If a third perturbation is made along the same line, it can be shown that the C matrix will not have full row rank, and, therefore, the third perturbation does not provide any extra useful information for the quadratic model. It should be noted, however, that the fact that the mixed term coefficients become zero is due to the maximally flat interpolation, and not an assumption.

In a situation where the mixed terms are important, this approach can easily be modified by introducing an appropriate transformation of variables. In such a case the perturbations can be carried out along the lines not necessarily parallel to the axes. The proposed fixed pattern of base points can thus be generalized while preserving the main advantages of our approach.

Theoretically speaking, the efficiency and simplicity of our models are achieved at the expense of some model accuracy. It should be stressed, however, that even so-called exact circuit simulation carries certain approximation of the actual physical behavior. Therefore, our approach provides an excellent modeling technique for many practical problems, especially when a very high accuracy is not really necessary. The method is suitable for up to $2n+1$ base points. It takes advantage of the concept of maximally flat quadratic interpolation and thus any number of base points between $n+1$ and $2n+1$ can be used.

IV. Computational Efficiency

A dynamic updating scheme was proposed in [4], which allows the existing model to be revised when a new base point is added. However, this simple updating may not be suitable if some of the base points are far from the region of interest. To maintain accuracy, it may be desirable to disregard such points. Our method can be used to rebuild the model very efficiently whenever it is needed. However, it can also be used within the concept of dynamic updating provided that the base points are selected in the aforementioned manner and their number does not exceed $2n+1$.

In this section we compare computational efficiency of our approach with that of the original method of [4]. To unify the comparison we assume that exactly $2n+1$ base points are used to build the model. In our approach the required storage is reduced to a minimum. Only $2n$ perturbations and $2n+1$ function values are to be stored. No matrix manipulations are needed. All calculations are simplified to (14) and (15). The operational count to calculate all coefficients using this pattern can be merely $4n$. *In this new approach, the storage requirement and computational count vary linearly with the number of variables.* For the original method [4], at least, all base points, matrices $\boldsymbol{Q}_{11}$ and $\boldsymbol{C}$ are stored in three arrays with dimensions $n\times(2n+1)$, $(n+1)\times(n+1)$ and $n\times n\times(n+1)/2$, respectively, and the computational count is $O(n^4)$. *For the original approach, the storage requirement and computational count vary cubically and quartically, respectively, with the number of variables.* For a circuit with 50 elements and m chosen as 101, we need storage consisting of two arrays of 101 and 100, and computational effort of 200 multiplications. The original approach would require storage consisting of three arrays of 50 by 101, 51 by 51 and 50 by 1275, respectively, and incomparable computational effort.

V. Design Centering Overview

Due to various kinds of fluctuations inherent in the manufacturing process, the circuit outcomes will present variations of responses from one another. Manufacturing yield is simply the ratio

$$N_{\text{pass}}/N_t$$

where N_{pass} is the number of circuit outcomes meeting the design specifications and N_t is the total number of circuit outcomes. Yield optimization simultaneously considers many circuits and takes a variety of fluctuations into account, aiming at a circuit design with satisfactory yield.

Suppose that we are given a nominal design $\boldsymbol{x}^0$ and predetermined statistics related to the tolerances of circuit parameters. We can generate some Monte Carlo outcomes, $\boldsymbol{x}^i$, $i=1,2,\cdots,K$, according to the statistical distribution of the toleranced circuit parameters. Each circuit outcome is associated with an acceptance index given by

$$I_a(\boldsymbol{x}^k)=\begin{cases}1, & \text{if } \boldsymbol{x}^k \text{ satisfies the specifications}\\ 0, & \text{otherwise.}\end{cases} \tag{16}$$

Then, yield is estimated by

$$Y\simeq\left[\sum_{k=1}^{K} I_a(\boldsymbol{x}^k)\right]\Big/K. \tag{17}$$

Consequently, our problem of circuit optimization is stated as

$$\underset{\boldsymbol{x}^0}{\text{maximize}}\ Y.$$

A number of algorithms have been proposed for statistical design centering, e.g., by Director and Hachtel [5] (the simplicial approximation), Soin and Spence [6] (the center of gravity method), Bandler and Abdel-Malek [1], [2] (updated approximations and cuts), Styblinski and Ruszczynski [7] (stochastic approximation), Polak and Sangiovanni-Vincentelli [8] (outer approximation), and Singhal and Pinel [9] (parametric sampling). Here we briefly review the approach employing the one-sided l_1 optimization technique proposed by Bandler and Chen [10] and used in our examples.

Simulation of each circuit outcome determines a set of error functions,

$$\boldsymbol{e}(\boldsymbol{x}^i)=\left[e_1(\boldsymbol{x}^i)\quad e_2(\boldsymbol{x}^i)\quad\cdots\quad e_M(\boldsymbol{x}^i)\right]^T \tag{18}$$

where M is the number of specifications applied. Then, the generalized l_1 function $v(\boldsymbol{e}(\boldsymbol{x}^i))$ can be calculated from $\boldsymbol{e}(\boldsymbol{x}^i)$. Consider the one-sided l_1 objective function for yield optimization [10] defined by

$$u(\boldsymbol{x}^0)=\sum_{i\in J}\alpha_i v(\boldsymbol{e}(\boldsymbol{x}^i)) \tag{19}$$

where $J=\{i|v(\boldsymbol{e}(\boldsymbol{x}^i))>0,\ i=1,2,\cdots,N\}$ and α_i are properly chosen nonzero multipliers. Using the optimization algorithm in [10] to minimize $u(\boldsymbol{x}^0)$, we can achieve a centered design with improved yield. This design centering approach is used in the two examples presented in the following section.

Statistical design centering involves a very large number of circuit simulations and, therefore, it is essential to reduce the CPU time needed for simulation. In such a design, the trend of the circuit response hypersurface seems to be more important than the accuracy of the individual circuit responses. Therefore, our approximation technique is capable of enormous acceleration of the design process without substantial loss of accuracy in the vicinity of a nominal circuit. Once the approximate model is established it is evaluated at all statistical outcomes that are sampled according to the statistical distribution of interest. There is no limit to the type of distribution, correlation of the variables, etc., provided that the statistical samples do not fall too far away from the region of approximation.

VI. Examples

A. Design of an 11-Element Low-Pass Filter [9], [11]

We have tested our quadratic approximation method within the framework of a circuit design program, which uses a general-purpose simulator and the generalized l_1 centering approach outlined in the preceding section. For the quadratic approximation, the individual circuit responses are chosen as the functions to be approximated. The reference point is defined as the nominal point. At each iteration, a set of quadratic models is built. The models are evaluated for all outcomes, i.e., the statistically sampled circuits. The objective function is calculated from the resulting approximate error functions.

A low-pass filter with 11 elements [9], [11], shown in Fig. 1, was used in this example. The upper specification was 0.32 dB at (0.02,0.04,$\cdots$,1 Hz), and the lower specification 52 dB at 1.3 Hz, on the insertion loss. A tolerance of 1.5 percent was assumed for

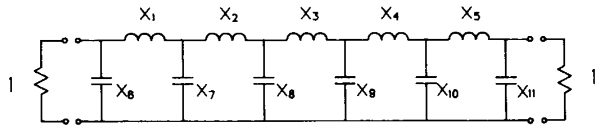

Fig. 1. The *LC* low-pass filter [9], [11].

TABLE I
COMPARISON OF STATISTICAL DESIGN OF A LOW-PASS FILTER WITH AND WITHOUT QUADRATIC APPROXIMATION

Component	Nominal Design	Exact Simulation		Quadratic Approximation	
		Phase 1	Phase 2	Phase 1	Phase 2
x^i	x^0	x^1	x^2	x^3	x^4
x_1	0.22510	0.22572	0.22512	0.22266	0.21669
x_2	0.24940	0.24903	0.24944	0.25045	0.25131
x_3	0.25230	0.25269	0.25276	0.25268	0.25083
x_4	0.24940	0.24908	0.24882	0.25028	0.24067
x_5	0.22510	0.22568	0.22594	0.22335	0.22120
x_6	0.21490	0.21589	0.21658	0.22163	0.23347
x_7	0.36360	0.36313	0.36275	0.36291	0.37008
x_8	0.37610	0.37625	0.37698	0.37938	0.37217
x_9	0.37610	0.37633	0.37561	0.37156	0.38529
x_{10}	0.36360	0.36313	0.36305	0.36226	0.37232
x_{11}	0.21490	0.21587	0.21674	0.22168	0.21893
Yield Estimated from Exact Simulation *	54.0%	61.7%	63.7%	70.2%	79.7%
Yield Estimated from Quadratic Approximation **	54.0%			74.0%	84.5%
Number of Outcomes Used for Optimization		200	200	200	200
Starting Point		x^0	x^1	x^0	x^3
Number of Simulations		48000	31200	529	828
Number of Iterations		9	7	10	19
CPU Time (VAX 8600)		96.2min.+	62.4min.+	2.5min.	3.9min.
CPU Time (MicroVAX)		481min.	312min.	12.3min.	19.5min.+

CPU times do not include yield estimation based on actual simulation.
* The yield is estimated using 1000 outcomes.
** The yield is estimated using 200 outcomes used in design.
+ The CPU time is approximately given by assuming that the speed ratio of VAX 8600 to MicroVAX is 5.

all elements. Outcomes were uniformly distributed between tolerance extremes. The starting point was the result of a synthesis procedure [9].

Results and comparisons are given in Table I. Two designs within the same optimization environment were carried out. The only difference between the two approaches was the way of calculating the circuit responses. The actual circuit simulations were used in the first design and our approximation method was utilized in the second one. Each design consisted of two successive centering processes shown as phases 1 and 2 in Table I. Two phases of design with actual simulations took approximately 96.2 and 62.4 min on VAX 8600 and required 48000 and 31200 circuit simulations, respectively. Achieved yields were 61.7 and 63.7 percent, respectively. Two phases of design with our approximation method used only 2.5 and 3.9 min on the VAX 8600. Only 529 and 828 actual circuit simulations were required. At the solutions of the two phases, the final yields were 70.2 and 79.7 percent, respectively. In all cases the yield values were estimated from 1000 Monte Carlo samples using exact circuit simulations. For the purpose of clarity, the CPU time needed for this yield estimation is not included in the aforementioned CPU times.

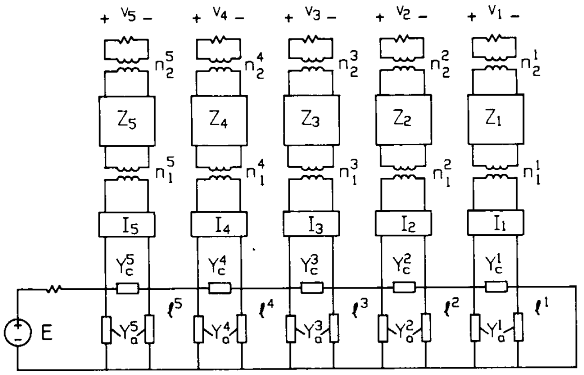

Fig. 2. The equivalent circuit of a 5-channel multiplexer. In the ith channel, V_i is the output, Z_i denotes the multi-coupled cavity filter, I_i symbolizes the impedance inverter, n^i_1 and n^i_2 are the input and output transformer ratios, Y^i_c and Y^i_a represent the nonideal series junction susceptances, and l^i stands for the waveguide spacings.

Interestingly, for this example, our method not only presented greatly reduced computational effort as compared to the actual simulation approach, but also reached a higher final yield. It suggests that accurate, time-consuming exact circuit simulation does not necessarily result in a better final yield. This statement, however, cannot be generalized any further.

While not illustrated in the table, we have used this example to compare the efficiency of our method w.r.t. the original maximally flat approach [4]. Using the same base points and employing the same scheme of rebuilding the models at each iteration, the original approach required approximately 5.3 and 8.3 min for the same two phases that our method took 2.5 and 3.9 min to finish. It should be noted that in both cases the CPU time needed to build and/or to evaluate the model constitutes only a fraction of the overall time, thus the remaining portions are common for the two approaches.

B. Design of a 5-Channel Microwave Multiplexer

This example is a 5-channel 12-GHz contiguous band microwave multiplexer consisting of multi-cavity filters distributed along a waveguide manifold [12]. Fig. 2 illustrates the equivalent circuit of the multiplexer. Tuning is essential and expensive for multiplexers to satisfy the ultimate specifications. The goal of this design is to easy the tuning process.

In order to take the appropriate tolerances into account, specifications were chosen to be 10 dB for the common port return loss and for the individual channel stopband insertion losses, resulting in 124 nonlinear constraint functions. Design variables included 60 couplings, 10 input and output transformer ratios, and 5 waveguide spacings. Tolerances of 5 percent were assumed for the spacings, and tolerances of 0.5 percent for the remaining variables. The starting point was the solution of the conventional minimax nominal design w.r.t. specifications of 20 dB. The corresponding responses are shown in Fig. 3. The estimated yield w.r.t. specifications of 10 dB at this point was 75 percent.

Yield optimization was carried out on the CRAY X-MP/22 using the generalized l_1 centering algorithm [10], our approximation scheme and utilizing a multiplexer simulation program [12]. The process consisted of 4 phases as shown in Table II. At the beginning of each phase, a set of quadratic models corresponding to 124 responses was constructed. These models were used for all outcomes in the phase. This is different from the first example

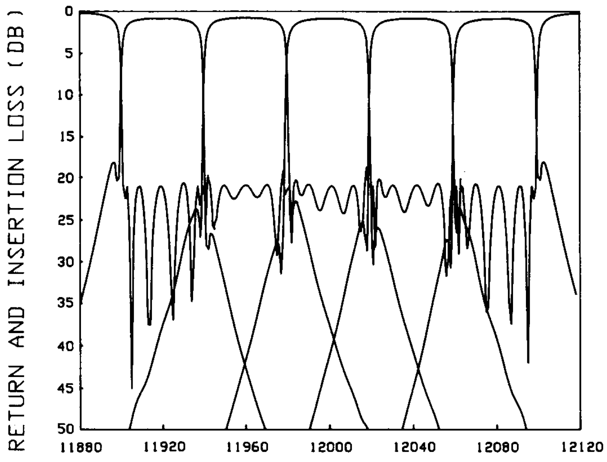

Fig. 3. Optimized return and insertion loss versus frequency for the 5-channel multiplexer.

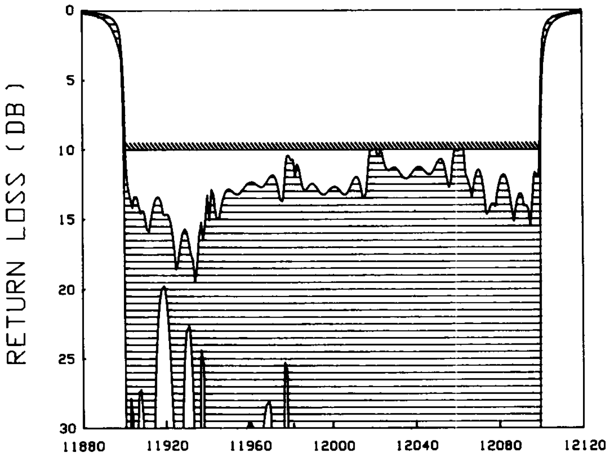

Fig. 5. The envelope containing return loss responses of acceptable circuits among 3000 Monte Carlo samples.

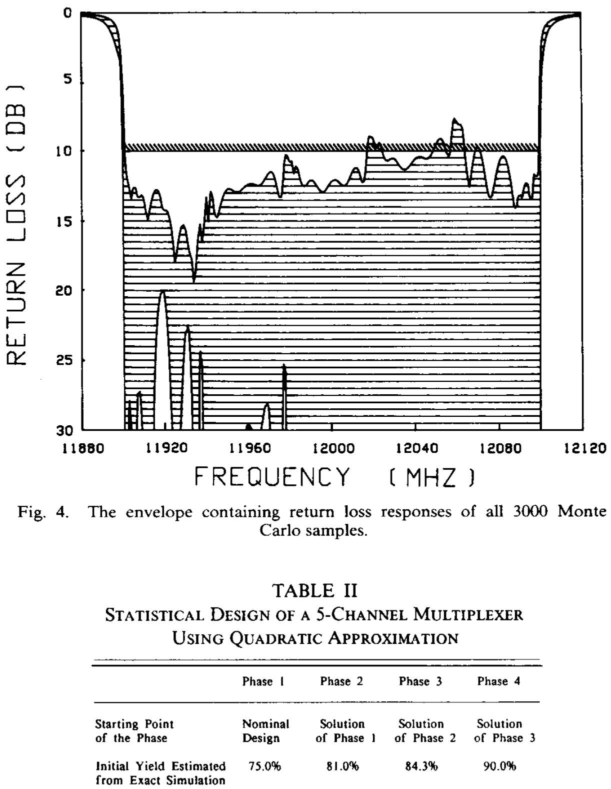

Fig. 4. The envelope containing return loss responses of all 3000 Monte Carlo samples.

TABLE II
STATISTICAL DESIGN OF A 5-CHANNEL MULTIPLEXER USING QUADRATIC APPROXIMATION

	Phase 1	Phase 2	Phase 3	Phase 4
Starting Point of the Phase	Nominal Design	Solution of Phase 1	Solution of Phase 2	Solution of Phase 3
Initial Yield Estimated from Exact Simulation	75.0%	81.0%	84.3%	90.0%
Initial Yield Estimated from Approximation	56.3%	69.0%	69.3%	92.0%
Number of Outcomes Used for Optimization	50	100	150	200
Number of Iterations	4	6	6	4
Final Yield Estimated by Simulation	81.0%	84.3%	90.0%	90.3%
Final Yield Estimated by Approximation	77.3%	77.3%	91.3%	94.0%
CPU Time (CRAY X-MP/22)	16.5s	17.6s	17.8s	17.6s

CPU times do not include yield estimation based on actual simulation. All yields are estimated using 300 samples.

where the quadratic models were rebuilt at each optimization iteration.

Four phases took totally 69.5 s on the CRAY X-MP/22 to reach a 90-percent estimated yield. This approach allowed us to handle this large optimization problem (with 75 toleranced variables, 124 constraints, and up to 200 statistically perturbed circuits) in acceptable CPU time. At the solution, the return losses of 3000 outcomes and of satisfactory outcomes are contained within the envelopes in Figs. 4 and 5, respectively.

VII. Conclusions

In this paper we have presented a highly efficient quadratic approximation technique. The new approach takes advantage of the maximally flat interpolation and of a fixed pattern of base points, thus substantially reducing the computational effort and required storage. A set of extremely simple formulas to calculate model coefficients has been derived. The elegance of this approach is its conciseness and applicability. The very strong impact of our approach on the feasibility of statistical design of larger circuits should not be underestimated. From the results of two examples of statistical design, very high efficiency and the capability of handling large problems have been proven. It should also be noted that our approach is suitable for a large variety of applications where a large number of expensive simulations is involved.

Acknowledgment

The authors wish to thank Dr. S. H. Chen and M. L. Renault of Optimization Systems Associates Inc., Dundas, Ont., Canada, for providing some necessary software and numerical results of the examples. Comments and suggestions from the reviewers are also appreciated.

References

[1] J. W. Bandler and H. L. Abdel-Malek, "Optimal centering, tolerancing, and yield determination via updated approximations and cuts," *IEEE Trans. Circuits Syst.*, vol. CAS-25, pp. 853–871, 1978.

[2] H. L. Abdel-Malek and J. W. Bandler, "Yield optimization for arbitrary statistical distributions: Part I—Theory," *IEEE Trans. Circuits Syst.*, vol. CAS-27, pp. 245–253, 1980.

[3] D. E. Hocevar, M. R. Lightner, and T. N. Trick, "Monte Carlo based yield maximization with quadratic model," in *Proc. IEEE Int. Symp. Circuits and Systems*, Newport Beach, CA, pp. 550–553, 1983.

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

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

[6] R. S. Soin and R. Spence, "Statistical exploration approach to design centering," *Proc. Inst. Elec. Eng.*, vol. 127, pt. G., pp. 260-269, 1980.

[7] M. A. Styblinski and A. Ruszczynski, "Stochastic approximation approach to statistical circuit design," *Electron. Letters*, vol. 19, no. 8, pp. 300-302, 1980.

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

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

[10] J. W. Bandler and S. H. Chen, "Circuit optimization: the state of the art," *IEEE Trans. Microwave Theory Tech.*, vol. MTT-36, pp. 424-443, 1988.

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

[12] 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.

Identification Via Fourier Series for a Class of Lumped and Distributed Parameter Systems

B. M. MOHAN AND K. B. DATTA

***Abstract*—Operational matrix of integration, as well as one shot operational matrix for repeated integration (OSOMRI) is used in this paper to estimate the parameters, and initial and boundary conditions of linear time-invariant (LTI) lumped parameter systems. It is demonstrated that OSOMRI provides better accuracy over conventional operational matrix of integration. Moreover, an algorithm for distributed parameter system identification via Fourier series is also included. Finally, a comparative study of the estimates obtained by the proposed method for both the systems with those available in the literature by other methods is also carried out.**

I. Introduction

Identification of lumped parameter systems (LPS) via Block-Pulse and Walsh functions [2], Chebyshev first and second kinds, Legendre and Jacobi, Laguerre and Hermite ([1] and the references therein) has already been studied. Similarly, others have used various orthogonal functions [5] apart from Walsh functions [3] for distributed parameter systems (DPS) identification. Inspite of these large number of contributions, it appears that no concerted effort has been made to explore the potentiality of Fourier series in system identification which consequently forms the main objective of this paper.

In the context of analysis of linear time-invariant (LTI) systems, Paraskevopoulos *et al.* [4] have already developed an integration operational matrix for Fourier series. In this paper, a more general operational matrix, with a merit that it can be applied over any arbitrary finite interval of integration, is developed. In order to improve the accuracy of operational matrix for repeated integration of Fourier basis vector the concept of OSOMRI, originally introduced by Rao and Palanisamy [2] in connection with Walsh functions, is introduced. Essentially based on this OSOMRI, an algorithm for the identification of LTI single-input single-output (SISO) continuous LPS is presented in addition to an algorithm for the identification of first-order continuous DPS in this brief. Numerical examples with detailed comparison of results obtained via proposed Fourier series method and other existing methods are provided.

II. Preliminaries of Fourier Series

The Fourier series or Fourier expansion corresponding to a function $f(t)$, which satisfies Dirichlet conditions, is approximately given by

$$f(t) \simeq f_0\phi_0(t) + \sum_{j=1}^{n-1}\left[f_j\phi_j(t) + f_j{}^*\phi_j^*(t)\right] = \boldsymbol{f}^T\boldsymbol{\phi}(t) \quad (1)$$

where the Fourier coefficients f_0, f_j and $f_j{}^*$ are

$$\left.\begin{aligned} f_0 &= \alpha\int_{t_i}^{t_f} f(t)\,dt \\ f_j &= 2\alpha\int_{t_i}^{t_f} f(t)\,\phi_j(t)\,dt;\ f_j{}^* = 2\alpha\int_{t_i}^{t_f} f(t)\,\phi_j^*(t)\,dt \end{aligned}\right\} \quad (2)$$

$$\left.\begin{aligned} \phi_j\,(t) &= \cos\left\{j\pi\left[2t-(t_i+t_f)\right]\alpha\right\}, \qquad j=0,1,2,\cdots \\ \phi_j^*(t) &= \sin\left\{j\pi\left[2t-(t_i+t_f)\right]\alpha\right\}, \qquad j=1,2,3,\cdots \end{aligned}\right\} \quad (3)$$

t_i – initial t; t_f – final t; $\alpha = 1/(t_f - t_i)$

$$\boldsymbol{f} = \left[f_0, f_1, \cdots, f_{n-1}, f_1^*, \cdots, f_{n-1}^*\right]^T \quad (4)$$

and

$$\boldsymbol{\phi}(t) = \left[\phi_0(t), \phi_1(t), \cdots, \phi_{n-1}(t), \phi_1^*(t), \cdots, \phi_{n-1}^*(t)\right]^T \quad (5)$$

A. Operational Matrix for Integration of Fourier Basis Vector $\boldsymbol{\phi}(t)$

Operational matrix for integration of $\boldsymbol{\phi}(t)$, denoted by E, may be obtained from the following steps.

i) Integrate every element of $\boldsymbol{\phi}(t)$ with respect to t,
ii) Express the result of step (i) in truncated Fourier series,
iii) Put the result of step (ii) in a vector–matrix form to obtain

$$\int_{t_i}^{t}\boldsymbol{\phi}(\tau)\,d\tau \simeq E\boldsymbol{\phi}(t) \quad (6)$$

where E is a nonsingular $(2n-1)\times(2n-1)$ matrix having the form

$$E = (t_f - t_i)\begin{bmatrix} 1/2 & \boldsymbol{0}^T & \boldsymbol{w}^T/\pi \\ \boldsymbol{0} & 0 & S/2\pi \\ -\boldsymbol{w}/2\pi & -S/2\pi & 0 \end{bmatrix}$$

with

$$\boldsymbol{w} = \left[1, -1/2, 1/3, \cdots, (-1)^n/(n-1)\right]^T$$

Manuscript received March 10, 1987; revised April 23, 1988 and February 16, 1989. This paper was recommended by Associate Editor R. Liu.

B. M. Mohan was with the Indian Institute of Technology, Kharagpur, India. He is now with the Department of Electrical and Electronics Engineering, Regional Engineering College, Tiruchirapalli 620015, India.

K. B. Datta is with the Department of Electrical Engineering, Indian Institute of Technology, Kharagpur 721 302, India.

IEEE Log Number 8929761.

0098-4094/89/1100-1454$01.00 ©1989 IEEE