Extraction of the Parameters of Equivalent Circuits of Microwave Transistors Using Tree Annealing

GRIFF L. BILBRO, MEMBER, IEEE, MICHAEL B. STEER, MEMBER, IEEE,
ROBERT J. TREW, SENIOR MEMBER, IEEE, CHAO-REN CHANG,
AND STEVEN G. SKAGGS

Abstract — The problem of extracting a physically based equivalent circuit model for a HBT transistor from S-parameter measurements is solved with a new formulation of simulated annealing. The physical model necessary for an accurate representation of the HBT leads to an extraction problem with many local minima. A satisfactory minimum can be found by conventional gradient-based techniques only with considerable expert guidance. The proposed algorithm finds equivalent circuits as good as those from conventional techniques but without human intervention. It is more efficient than the conventional stochastic simulated annealing because it accumulates a probability density of good equivalent circuits which it subsequently uses to refine its statistical search for the best equivalent circuit.

I. INTRODUCTION

As quantitative accuracy in the design phase becomes critical, it becomes increasingly important to develop accurate transistor equivalent circuit models. Although the equivalent circuit characterization technique is widely used, the lack of a general-purpose global optimization scheme has prevented the application of realistic, physically based equivalent circuits and the incorporation of physical insight into the optimization process.

Gradient descent algorithms are most commonly used in parameter extraction, but even when these algorithms are restarted from several randomly chosen initial points, results are seldom satisfactory. Measurement error coupled with the large number of elements of a physically based equivalent circuit leads to an error function (the difference between the S parameters calculated for the equivalent circuit and those measured) with many non-physical local minima in addition to the global minimum. The number of additional minima grows rapidly as the number of elements of the equivalent circuit is increased, and gradient-based algorithms generally terminate in one of these. Expert consideration of the equivalent circuit at a local minimum with respect to the actual physics of the device can be used to estimate a superior starting point for another gradient-based optimization. This process is painstaking but can result in realistic equivalent circuits for the HBT transistor.

An alternative to gradient descent has recently been proposed by Vai et al. [1], who used simulated annealing (SA), a near global optimization technique, for the parameter extraction of an eight-element equivalent circuit. In SA, a current equivalent circuit is perturbed and the new error for the candidate is calculated. If its error is smaller, this candidate circuit replaces the current circuit as in a descent-algorithm-based parameter extractor. But, sometimes, in distinction to descent-based procedures, a candidate with a larger error may be accepted in accordance with a precise probabilistic criterion which becomes less tolerant of “bad” moves at later stages of the algorithm. The success of this approach depends on generating moves that are neither always accepted nor always rejected [2]. Designing a generator that maintains this balance can become more difficult as the dimension of the error function increases: for continuous optimization this difficulty can become insurmountable at five to ten dimensions [3]. This difficulty arises when the error function is “highly anisotropic” in the vicinity of important minima [3]. Another difficulty with the perturbative approach to simulated annealing is when the error function is “too irregular” and has many deep local minima, or “gopher holes,” as they have been called in the literature of circuits and devices [4]. Many circuit optimization problems, including parameter extraction, have error functions with many gopher holes and so are ill suited to perturbative simulated annealing [4].

We report here a parameter extraction procedure which tolerates gopher holes. As such it can be adapted to many circuit optimization problems. However it was specifically developed for the extraction from measurements of the parameters of microwave transistors. This is one of the most difficult optimization problems, as the measured data are corrupted by measurement uncertainties and the assumed lumped element equivalent circuit can only ap-
proximately model an actual active device. The algorithm is a variation of simulated annealing and uses a familiar data structure (a binary tree) to record previous moves and uses this information to optimally generate subsequent candidates. This information optimally represents both anisotropy and irregularity when it is practical to collect this information by sampling the error function. Because of the binary tree, we call the algorithm tree annealing (TA).

Because of the novelty of simulated annealing in microwave engineering, we begin our discussion with an overview of this technique and a qualitative comparison with gradient descent optimization algorithms. We then present our tree annealing theory, followed by the extraction of the parameters of a HBT using a physically based equivalent circuit model and measured data from 45 MHz to 26.5 GHz.

II. SIMULATED ANNEALING IN COMBINATORIAL AND CONTINUOUS OPTIMIZATION

Simulated annealing was introduced by Kirkpatrick et al. [5] in 1983 and is a smart random search technique which is often more efficient than exhaustive search yet more robust than gradient descent. Its behavior is controlled by an externally specified parameter, usually called temperature, $T$, with the same units as the error function. When $T$ is larger than the extreme range (maximum to minimum) of the error function, SA explores the entire parameter space using a uniformly generated random walk and with no preference for lower error. As $T$ gets smaller, this undirected exploration changes. When $T$ is small enough, SA becomes a descent algorithm. At intermediate values of $T$, SA moderates its behavior between these two extremes: it is random but “prefers” lower error in a precise way. By gradually decreasing $T$, the search is systematically concentrated into regions likely to contain a global minimum, but is still random enough to escape local minima. In some cases, convergence to the global minimum is provable [6], [7].

Instead of directly minimizing the objective function $E(x)$, Kirkpatrick et al. considered maximizing the related probability density, $p_f(x) \propto \exp(-E(x)/T)$. Since the exponential function is monotonic, this does not change the problem and would not facilitate its solution except that $p_f$ is now amenable to the powerful Metropolis sampling procedure [8] (described later in this section). The modes of $p_f$ coincide with the minima of $E$ at all $T$, and at low enough values of $T$, $p_f(x)$ is so sharply peaked at the lowest values of $E(x)$ that virtually no other points contribute to the distribution. The normalization factor is a function of $T$.) Unfortunately the convergence time of the Metropolis procedure can be exponentially long for low $T$. Kirkpatrick et al. reported a solution in 1983. They observed that large crystals could be grown most rapidly by gradually cooling a melt. Since the Metropolis procedure simulates the same physical statistics that govern crystal growth, they suggested that sampling at low $T$ equilibrium could be done most rapidly by “annealing” or gradually reducing $T$ from a high initial value. Thus, instead of directly minimizing the error, SA maximizes the probability of sampling low error states. This is often a much easier problem to solve.

SA, as well as TA, is understood in terms of averages of random processes instead of a single efficient descent down a locally optimal trajectory. In combinatorial optimization, even the Lin–Kernighan algorithm, a sophisticated descent procedure with backup, has been described as “bottom-crawling” compared with SA, which was contrasted as “floating across the upper or middle regions of the larger valleys” [9]. Only gradually does this average level float down toward the lowest values of the objective.

These concepts are illustrated in Fig. 1, where the error on a one-dimensional slice through a higher dimensional parameter space is plotted. A descent procedure would terminate in one of the three minima, depending on initialization. SA concentrates its search on points that have an error value that is roughly proportional to a power of the temperature. So SA, while seeking a good solution, encircles each minimum with samples having comparable errors. In Fig. 1, these rings are indicated for a sequence of four temperatures: $T_1 > T_2 > T_3 > T_4$. As the temperature is reduced each shell contracts and by $T_2$ has split into two rings, each centered at a local minimum. Below $T_3$, the ring around the shallower valley on the left has become depopulated, leaving only states surrounding the lowest minimum. In this way, shallow valleys tend to be eventually discarded but not before the sides of the valley are explored by the shrinking ring. SA is effective when deep minima have wide rims. SA fails when the global minimum occurs in a gopher hole, which is too steep and narrow to be discovered by sampling at higher temperatures [4].

SA is widely used in combinatorial optimization but less commonly in continuous optimization [2]. However, Vai et al. have recently applied SA to the continuous optimization problem of estimating the parameter values.
of a seven- or eight-element equivalent circuit for a MODFET [1]. In SA a current state $x$ is randomly perturbed into a candidate state $y$. The candidate is either accepted or rejected by the Metropolis criterion [8], that is, with the probability of accepting $y$ given $x$ of

$$a(y|x) = \begin{cases} 1 & \text{if } p_T(y) > p_T(x) \\ \frac{p_T(y)}{p_T(x)} & \text{otherwise.} \end{cases}$$

That is,

$$a(y|x) = \min(1, \frac{p_T(y)}{p_T(x)}). \tag{1}$$

If the candidate is accepted, it replaces the current state in the next time step; otherwise the current state persists and the candidate is discarded. The resulting sequence of states is guaranteed to asymptotically sample the density $p_T$.

Vanderbilt and Louie [3] have reported that SA generally becomes inefficient for continuous optimization as the number of unknowns increases. The principal complication is that the optimal magnitude and direction of steps are not known in advance. Steps that are too small result in unacceptably slow optimization. However, if steps are too large, important minima may not be detected as the search procedure almost always finds candidates with unacceptably large errors. If the minimum of the error function lies in a highly anisotropic valley, the straightforward algorithm suffers the worst of both inefficiencies. As the dimension of the search increases, the effect of this anisotropy worsens. Vanderbilt and Louie improved the efficiency of their version of SA enough to optimize functions of ten continuous unknowns by keeping track of the parameter covariances. However Vanderbilt and Louie report some persistent difficulties with deep isolated minima. We have experienced numerical difficulties manipulating the covariance matrices when the number of dimensions is higher than about 10, whereas physically based equivalent circuits require more than ten elements.

III. TREE ANNEALING THEORY

We have developed a reformulation of SA that can robustly extract a 14-parameter transistor equivalent circuit from data. Our reformulation generates candidates from a global approximation of $p_T(x)$ rather than by perturbation of the current state. We call this global approximation $g(x)$ because it is used to generate candidates. We call our algorithm tree annealing because we maintain $g$ in a multidimensional binary tree (or more precisely, a k-d tree data structure [10]) which also allows us to conveniently sample from $g$. This can be compared to Vanderbilt and Louie’s multivariate Gaussian model, which they used to accumulate a distribution and to subsequently sample it. However our tree-based algorithm is superior in two ways. One is that it makes weaker assumptions about the error function which can now have several isolated minima without compromising the underlying quadratic model in Vanderbilt and Louie. The other is that we employ a powerful modification of the Metropolis acceptance criterion which permits optimal and unrestricted exploitation of any approximation to the error function. Like the familiar Metropolis procedure, ours can be shown [11] to asymptotically sample $p_T(x)$ but it requires an unfamiliar form of the Metropolis criterion [12].

Like Vanderbilt and Louie’s Gaussian model, our tree is used both to record the density of previous samples and also to generate candidates for future samples consistent with that recorded density. We call the recorded density $g(x)$ and the correct density $p_T(x)$. The density $g$ is not assumed to coincide with $p_T$; in fact, the recorded density changes slowly with decreasing temperature so that it is always slightly incorrect for the current temperature. However if care is taken to ensure that $g$ never vanishes, any error in $g$ can be rigorously compensated for by a modification of the Metropolis acceptance probability [11].

In the usual Metropolis procedure, candidates are sampled from a uniform distribution $g$ and these samples are converted to samples of $p_T$ by accepting each with probability (1) and rejecting the rest. In our procedure, $g$ is not uniform, but samples from it can be converted to samples from the true $p_T$ by accepting each $y$ with probability

$$a(y|x) = \min \left( 1, \frac{g(x)p_T(y)}{g(y)p_T(x)} \right) \tag{2}$$

where $x$ is the current state. This corrects the familiar ratio of desired densities, $p_T(y)/p_T(x)$ in (1) by dividing out the ratio $g(y)/g(x)$ where the values of $g(y)$ and $g(x)$ are the numerical probabilities of generating $y$ or $x$. The ratio is the bias toward $y$ relative to $x$ and intuitively makes $y$ less likely to be accepted if it is more likely to be regenerated.

Convergence to the correct stationary distribution is guaranteed even when $g(x) \neq p_T(x)$, although efficiency is reduced by a poor $g$. Because of this correctness, we can accumulate the sampled $x$’s to improve a poor $g$ and raise subsequent efficiency. Furthermore, in the final stages of the algorithm, these samples can be used to estimate the precision with which the optimal values of $x$ have been determined.

The highest level flow of control in tree annealing can be seen in Fig. 2. After root is initialized at level zero with the limits $lo$ and $hi$ of the entire search space and $T$ is initialized, an initial candidate circuit $x$ is randomly drawn. The main loop is then entered and a candidate circuit $y$ is generated. The circuits $x$ and $y$ are compared probabilistically and $y$ is usually accepted as a better candidate if its fit error (defined as the sum of the squares of the differences between each $S$ parameter calculated for $x$ and those measured) does not exceed the fit error of $x$ by more than $T$. If $y$ is accepted, it overwrites $x$. Whichever circuit survives is used to adjust the tree. The temperature is then reduced slightly and the loop is repeated until $T$ is low enough (i.e., the fit error is below some specified tolerance).
Fig. 2. Top level flowchart of the TA algorithm.

The essential algorithmic difference between this procedure and the usual simulated annealing technique [7] is that our candidate $y$ is generated from $\text{root}$ rather than directly from the current state $x$. In our algorithm, $x$ influences generation only indirectly through the action of ADJUST TREE. Our acceptance of $y$ involves the entire tree in addition to the usual $y$, $x$, and $T$. This is because (2) requires the numerical values of $g$ at $x$ and $y$. These numerical values must of course agree with the actual probability of generating the corresponding points. This is easy to do with a binary tree. In each node of the tree, we maintain the total count of samples that historically originated in each half of the space assigned to that node, if any samples have originated from that particular node. If so, the node also maintains pointers to two children nodes which divide the space just as the counts are divided. In the generation process, we randomly chose a left or right child in accordance with the counts of previous samples. In the numerical evaluation of the probability for a point $y$, we retrace the path to $y$ and along the way accumulate the product of the probabilities from the root to the leaf containing $y$. Since it is a continuous density and not a discrete distribution, we must divide by the volume of the parameter space associated with the leaf containing $y$ (if the leaf is large, any particular point in it is less likely). Thus our modified Metropolis procedure samples $p_T(x)$ by probabilistically accepting samples generated from $g(x)$, which is kept in a tree.

IV. EXAMPLE

We will describe the operation of our TA algorithm for the simple one-dimensional problem of minimizing $E$ in Fig. 3(a). On the left-hand sides of parts (b)–(d) in Fig. 3, the solid curve is $p_T$ at temperature $T$, which here reduces with each addition to the tree. The shaded region indicates $g$, the current estimate of $p_T$. The right-hand sides show the growth of the tree. The heights of the bars on the left-hand side are the numerical probabilities discussed in the preceding paragraph. The goal is to find the global minimum of $E$ in the search space $S$, which here is the interval $(0, L)$ assigned to the root node of the tree. Initially, the root node is the only node in the tree and GENERATE returns an initial $x$. Then a first candidate
y (from a uniform distribution on (0, L)) is generated so that \( g(x) = g(y) = 1/L \) as in Fig. 3(b). Since \( T \) is initially high, \( p_T(x) \approx p_T(y) \) and (2) is nearly unity and almost any \( y \) is accepted as the new \( x \). The root node maintains counts of samples from each half of its parameter space, in this one-dimensional case the subintervals \((0, 1/2L)\) and \((1/2L, L)\).

As this process is repeated, \( T \) gradually falls low enough that many \( y \)'s from the high \( E \), i.e., low \( p_T \), regions of the search space are rejected by the Metropolis procedure when the current \( x \) is from the "better" side. This bias is recorded in the counters \( n_{\text{eff}} \) and \( n_{\text{avg}} \), which are proportional to the averages of \( p_T \) on the corresponding subintervals. At low enough \( T \), these rejections become wasteful.

Before the efficiency falls too low, further rejections can be reduced by allocating children for the leaf (which is initially also the root) as in the next panel, Fig. 3(c). Each child is assigned half of its parent's search and now GENERATE chooses the left half of \( S \) with probability \( p_{\text{left}} = n_{\text{eff}} / (n_{\text{left}} + n_{\text{avg}}) \) and otherwise the right half. At the new leaf, \( y \) is still drawn uniformly but from a smaller region, so that \( \frac{g(y)}{g(x)} \) for a sample on the left is \( 2p_{\text{left}} / L \). In the figure, \( g(y) \) is higher on the right so that the ratio \( g(x)/g(y) \) is no longer always unity, but by construction this ratio is, on average, the same bias of the true distribution so that the acceptance probability of (2) is, on the average, unity. So once again every \( y \) returned by GENERATE is usually accepted as the new \( x \), but now the \( y \)'s are more likely to come from the "better" side, as in Fig. 3(d).

As before, \( T \) eventually falls low enough to disturb this balance again, but, as before, the leaves have recorded the bias by this time. Each leaf that is visited frequently enough to justify further refinement divides its space between two children and thereafter randomly chooses between them with the current bias. Every Metropolis sample, whether it is a newly accepted candidate or a persistent previous circuit, is counted at every node on its path from root to leaf and used to keep biases current. As the algorithm progresses, the tree tends to grow into regions from which samples most frequently come. Each additional level of the tree doubles the resolution with which it can model the parameter search space \( S \). Each drop in \( T \) tightens the acceptance criterion and without this additional resolution would lead to more rejections and more wasted evaluations of the objective function. In higher dimensions, the tree grows and the counts adjust so as to reflect whatever anisotropies and irregularities are important in \( p_T \). Theoretically, for \( N \) dimensions the computation time is of the order of \( 2^{N^2} \) if an exhaustive search is required. However, in the problems we have looked at, the computation increase is better than quadratic (\( < O(N^2) \)), presumably because optimization problems tend to have unequal complexity in each dimension.

V. PARAMETER EXTRACTION

The TA optimization algorithm was used to extract the parameters of the HBT transistor of Mishra et al. [13] using the physically based equivalent circuit of Fig. 4 and de-embedded scattering (S) parameter measurements from 45 MHz to 26.5 GHz. In the transistor model the current gain

\[
A = \frac{A(0)}{1 + \frac{f}{f_c}} e^{-2\pi f c / \tau_c}
\]

where \( A(0) \) is the dc current gain, \( \tau_c \) is the carrier transit time through the base, \( f \) is frequency, and \( F \) accounts for the variation in the base width and is the frequency at which the magnitude of the internal current gain is down to 0.707 of its low-frequency value. \( F \) cannot be measured directly and is determined from the measured unity current gain frequency \( f_c \) and the time constants in the equivalent circuit as \( f = 1/(2\pi \tau_c) \), where \( \tau_c = \tau_c - (\tau_c + R_C C + R_C C + R_C C) \) and \( \tau_c = 1/(2\pi f_c) \).

The extracted model element values are listed in Table I together with the element bounds used. The calculated \( S \) parameters are virtually indistinguishable from those measured. A more sensitive comparison is given in Fig. 5, where the measured stability factor, \( k \), and maximum available gain, \( G_{\max} \), are compared with those evaluated from the equivalent circuit. The agreement is excellent, indicating an accurate parameter extraction.
Also given in Table I are the standard deviations (σ) of the optimized element parameters for a 10% variation of the S parameters. Small σ implies that the error function is sensitive to variations in this element and so the element value is determined to a tight tolerance. The provision of error estimates indicates the importance of an element to the objective function minimization and provides useful information to device and circuit designers. This is particularly important in the tolerancing of MMIC's as it directs effort toward the control of elements with small σ which have most effect on external characteristics.

Various combinations of the elements of a transistor equivalent circuit are closely correlated, as shown in Table II. In this table the cross-correlation coefficients of the optimized elements are shown for a 10% variation from measured S parameters. These statistics are immediately available in the TA procedure. It can be seen that the dc current gain A(0) for R_L and R_WE are tightly correlated. Also most of the capacitors are tightly correlated. A high correlation coefficient indicates that the associated elements cannot be adequately resolved. If it is important to distinguish them, additional or independent measurements are required.

VI. DISCUSSION

The TA parameter extraction procedure as implemented by us requires that the upper and lower limits of the element values be specified by the user. Otherwise no human intervention is required. To date the TA parameter extraction procedure has never failed to identify the global minimum (as far as can be determined). In Table III the TA results are compared with multiple restart gradient descent data. Also shown are the mean square S-parameter errors, defined as

\[ \frac{1}{N} \sum_{\text{all frequencies}} |S_{ij}^C - S_{ij}^M|^2 \]

where the superscripts C and M refer to, respectively, the S parameters calculated for the equivalent circuit and to those measured; the frequency summation is from 0.45 MHz to 26.5 GHz calculated at 45 MHz intervals to 450 MHz and then from 500 MHz every 2 GHz; and N is the number of frequencies. The multiple restart gradient descent results were obtained by beginning with the previously optimized element values of the equivalent circuit of a similar device. Typically three to seven restarts were required to obtain a quality solution. Between each restart the physical merit of the extracted equivalent circuit was examined to determine the starting point for subsequent gradient optimization. Of course with more restarts we are more likely to find the global minimum but the high cost of human intervention practically limits the number of restarts that can be tried. Nevertheless, this approach proved to be the most successful of the commercially available gradient-descent-based approaches. (Application of Levenberg–Marquardt [16], [17] codes considerably improves the performance of gradient-based parameter extractors [18] although entrapment in local minima can still be a problem.) The robustness of the TA parameter extraction procedure is not free. TA is more computationally intensive than gradient-based algorithms and we have found that five to ten times more CPU time is required to extract equivalent circuits of comparable quality. However the major cost of gradient-based algorithms is the engineering time required to examine the extracted circuit, decide on its merits, and to choose physical starting points.

The distinctive features and heuristics of TA can be compared to those of SA and gradient descent algorithms as follows. Simple descent algorithms terminate at the first local minimum encountered from a particular starting point. If there is only one minimum or if the global minimum can be estimated, descent is the method of choice for optimization. If there are not too many local minima, descent algorithms can be conveniently restarted from many different (often randomly chosen) initial points and the best solution taken as the approximation of the global minimum. SA differs from descent algorithms by continually accepting some proportion of random moves up the error surface. SA is more efficient than random multistart descent algorithms if the local minima tend to be shallow and if the error surface is not dominated by deep winding grooves or valleys, so that a feasible proportion of uphill and downhill moves can be generated. TA goes a step further than SA by accumulating a record of accepted moves. Since the error function is generally smooth, good equivalent circuit candidates will tend to cluster, so that the neighborhood of an earlier good solution is an efficient place to randomly select a new candidate. Since there could be more than one cluster of good circuits and since the shape of the error function
TABLE II
Correlations of Calculated Equivalent Circuit Elements for 10% Fractional Errors in the Measured 3 Parameters

<table>
<thead>
<tr>
<th>LBB</th>
<th>RBB</th>
<th>CBB</th>
<th>RBB</th>
<th>RBB</th>
<th>CBB</th>
<th>EBB</th>
<th>REE</th>
<th>A(0)</th>
<th>CBB</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>-0.21</td>
<td>1.00</td>
<td>-0.50</td>
<td>0.15</td>
<td>1.00</td>
<td>-0.15</td>
<td>0.06</td>
<td>0.49</td>
<td>1.00</td>
</tr>
<tr>
<td>0.26</td>
<td>-0.28</td>
<td>0.02</td>
<td>0.23</td>
<td>1.00</td>
<td>0.35</td>
<td>0.07</td>
<td>-0.47</td>
<td>-0.24</td>
<td>-0.19</td>
</tr>
<tr>
<td>0.12</td>
<td>-0.08</td>
<td>0.02</td>
<td>0.04</td>
<td>0.12</td>
<td>0.10</td>
<td>-0.08</td>
<td>0.01</td>
<td>-0.23</td>
<td>0.03</td>
</tr>
<tr>
<td>0.15</td>
<td>0.14</td>
<td>0.01</td>
<td>0.02</td>
<td>0.12</td>
<td>0.10</td>
<td>-0.08</td>
<td>0.01</td>
<td>-0.23</td>
<td>0.03</td>
</tr>
<tr>
<td>0.17</td>
<td>-0.38</td>
<td>0.20</td>
<td>0.01</td>
<td>0.11</td>
<td>-0.20</td>
<td>0.40</td>
<td>-0.07</td>
<td>0.15</td>
<td>-0.01</td>
</tr>
<tr>
<td>0.46</td>
<td>0.46</td>
<td>0.14</td>
<td>0.07</td>
<td>0.05</td>
<td>0.12</td>
<td>-0.02</td>
<td>0.02</td>
<td>-0.01</td>
<td>0.03</td>
</tr>
<tr>
<td>0.07</td>
<td>-0.02</td>
<td>0.08</td>
<td>0.15</td>
<td>-0.04</td>
<td>0.10</td>
<td>0.15</td>
<td>0.14</td>
<td>0.09</td>
<td>1.00</td>
</tr>
<tr>
<td>0.32</td>
<td>0.32</td>
<td>0.02</td>
<td>0.01</td>
<td>0.03</td>
<td>0.01</td>
<td>0.03</td>
<td>0.01</td>
<td>0.03</td>
<td>1.00</td>
</tr>
<tr>
<td>0.10</td>
<td>0.10</td>
<td>0.02</td>
<td>0.03</td>
<td>0.09</td>
<td>0.10</td>
<td>0.10</td>
<td>0.10</td>
<td>0.10</td>
<td>1.00</td>
</tr>
<tr>
<td>0.32</td>
<td>0.32</td>
<td>0.02</td>
<td>0.03</td>
<td>0.09</td>
<td>0.10</td>
<td>0.10</td>
<td>0.10</td>
<td>0.10</td>
<td>1.00</td>
</tr>
</tbody>
</table>

TABLE III
Comparison of Optimized Parameter Values Obtained with Tree Annealing, Tree Annealing with Subsequent Gradient Optimization, and Gradient Optimization with Multiple Restarts

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DATA</th>
<th>ANNEALED DATA</th>
<th>MULTI-START GRADIENT DATA (GR)</th>
<th>ANNEALED / GRADIENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBB</td>
<td>41.4 pH</td>
<td>39.5 pH</td>
<td>1.06</td>
<td></td>
</tr>
<tr>
<td>RBB</td>
<td>50.2 Ω</td>
<td>58.3 Ω</td>
<td>1.17</td>
<td></td>
</tr>
<tr>
<td>CBB</td>
<td>575 fF</td>
<td>753 fF</td>
<td>1.02</td>
<td></td>
</tr>
<tr>
<td>RBB</td>
<td>18.0 Ω</td>
<td>18.3 Ω</td>
<td>0.99</td>
<td></td>
</tr>
<tr>
<td>RBB</td>
<td>2.11 Ω</td>
<td>2.24 Ω</td>
<td>1.04</td>
<td></td>
</tr>
<tr>
<td>CBB</td>
<td>72.0 Ω</td>
<td>77.1 Ω</td>
<td>1.03</td>
<td></td>
</tr>
<tr>
<td>RBB</td>
<td>2.73 Ω</td>
<td>2.44 Ω</td>
<td>1.12</td>
<td></td>
</tr>
<tr>
<td>A(0)</td>
<td>0.973</td>
<td>0.974</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>τC</td>
<td>1.07 ps</td>
<td>0.80 ps</td>
<td>1.34</td>
<td></td>
</tr>
<tr>
<td>CBB</td>
<td>24.7 Ω</td>
<td>31.4 Ω</td>
<td>1.28</td>
<td></td>
</tr>
<tr>
<td>RBB</td>
<td>30.8 kΩ</td>
<td>25 kΩ</td>
<td>1.23</td>
<td></td>
</tr>
<tr>
<td>RBB</td>
<td>3.70 Ω</td>
<td>3.88 Ω</td>
<td>1.05</td>
<td></td>
</tr>
<tr>
<td>LBB</td>
<td>63.2 pH</td>
<td>59.9 pH</td>
<td>1.06</td>
<td></td>
</tr>
<tr>
<td>CBB</td>
<td>37.4 fF</td>
<td>37.4 fF</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>ERROR</td>
<td>0.0220</td>
<td>0.0255</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Also shown is the ratio of the optimized values obtained with TA with subsequent gradient optimization and those obtained with multiple-restart gradient optimization. The error is the mean square-error of the scattering parameters from 0.045 MHz to 26.5 GHz calculated at 50 MHz intervals to 450 MHz and then every 2 GHz.

near the centers of the clusters is not necessarily spherical or isotropic, an enormous amount of information may have to be stored to record all of the random samples. However this information can be maintained efficiently in a binary tree and used efficiently by a modified Metropolis procedure. By storing this information many good solutions can be examined in the event that the global optimum is not physically correct. The preferred approach, however, is to incorporate additional physical insight into the error function. In the example presented here the measured unity current gain frequency is used with this effect.

Since we developed TA for the parameter extraction problem, we have run TA with a variety of cooling schedules and on other problems. We find its behavior similar to SA, except TA "jumps barriers" between deep minima and TA rejects fewer candidates at low temperatures.

VII. CONCLUSION

Good results were obtained from the parameter extraction technique, and the ability of tree annealing not to get trapped in local minima allowed a many-element physically based equivalent circuit model to be used. Tree annealing is essentially a smart random search technique and so requires many more functional evaluations than do gradient-based minimization algorithms. However, no user-supplied starting guess is required and the bounds on parameter values can be widely separated with little effect on optimization time. The initial trial solution is determined randomly within user-specified parameter bounds.

REFERENCES

Griff L. Bilbro (M'85) received the B.S. degree in 1973 from Case Western Reserve University in Cleveland, OH, and the Ph.D. degree in 1977 from the University of Illinois at Urbana-Champaign, where he was a National Science Foundation Graduate Fellow in physics.

He worked in industry until 1984, where he designed computer models of complex systems. He joined the Center for Communications and Signal Processing in the Department of Electrical and Computer Engineering at North Carolina State University as a Research Associate in 1984 to work on problems in image analysis. In 1988, he began developing practical engineering techniques for global optimization. He now participates in the High Frequency Electronics Laboratory at North Carolina State University and is interested in device physics, yield optimization, and parameter extraction for microwave circuits.

Chao-Ren Chang received the B.S. degree from National Cheng Kung University, Taiwan, Republic of China, in 1971 and the M.S. degree from North Carolina State University in 1980, both in electrical engineering.

From 1976 to 1979 and from 1981 to 1984, he was an assistant scientist in the Microwave Laboratory of the Chung Shan Institute of Science and Technology, Taiwan. Since 1984 he has been a teaching and research assistant at North Carolina State University, Raleigh, where he is currently completing the Ph.D. degree in electrical engineering. His research interests are in the areas of the computer-aided analysis, design and testing of nonlinear microwave circuits, and the characterization of microwave devices.

Michael B. Steer (S'76–M'82) received the Ph.D. degree in electrical engineering from the University of Queensland, Brisbane, Australia, in 1983 and is currently Assistant Professor of Electrical and Computer Engineering at North Carolina State University.

His research involves the simulation and computer-aided design of nonlinear analog circuits with large-signal excitation and of circuits with mixed analog and digital signals. He is currently working on the simulation of microwave analog circuits, delta-sigma modulators, high-speed printed circuit boards, and the computer-aided design of analog circuits using simulated annealing. Dr. Steer was named a Presidential Young Investigator in 1987.

Steven G. Skaggs received the B.S. degree in electrical engineering in 1989 from North Carolina State University, Raleigh. He is currently pursuing a master of science degree in electrical engineering and is a research assistant in the High Frequency Electronics Laboratory at North Carolina State University. His interests are in parameter extraction and in the computer-aided design of nonlinear microwave circuits using simulated annealing.